

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение
высшего образования

**Нижегородский государственный университет
им. Н.И. Лобачевского (ННГУ)**

Институт Информационных Технологий, Математики и Механики

Л.Г.Афраймович

**Информационные технологии в области принятия
решений. Часть 1**

Рекомендовано методической комиссией ИИТММ для студентов ННГУ, обучающихся по магистерской программе «Прикладная информатика в области принятия решений» направления подготовки «Прикладная информатика» 09.04.03.

Нижегород
2016 год

УДК 519.874

А-

А- Афраймович Л.Г., Учебно-методическая разработка «Информационные технологии в области принятия решений. Часть 1». – Нижний Новгород: Нижегородский госуниверситет, 2016. – 27с.

Рецензент: к.т.н., доцент Мееров И.Б.

В учебно-методической разработке приводится задание для лабораторных работ 1 и 2 спецсеминара «Информационные технологии в области принятия решений» магистров 1 курса направления подготовки «Прикладная информатика» 09.04.03, обучающихся по магистерской программе «Прикладная информатика в области принятия решений». Даются постановки прикладных задач, описываются подходы к решению, формулируется задание для самостоятельной работы и приводятся тестовые задачи, необходимые для выполнения заданий.

**© Нижегородский государственный
университет им. Н.И. Лобачевского, 2016
© Афраймович Л.Г.**

Оглавление

1. Лабораторная работа «Задача одномерного раскрыя»	4
1.1. Математическая модель	4
Содержательная постановка	4
Исходные параметры	4
Варьируемые параметры	4
Ограничения математической модели	4
Критерии оптимальности	4
Анализ сложности	5
1.2. Жадный алгоритм и итерационные стратегии	5
Жадный алгоритм	5
Итерационные стратегии	5
Стратегия 1	6
Стратегия 2	6
1.3. Задание	7
1.4. Тестовые задачи	7
2. Лабораторная работа «Задача о доставки заказов»	14
2.1. Математическая модель	14
Содержательная постановка	14
Исходные параметры	14
Варьируемые параметры	14
Ограничения математической модели	14
Критерии оптимальности	15
Анализ сложности	15
2.2. Метод ветвей и границ и стратегии ветвления, построения нижней и верхней оценок	15
Метод ветвей и границ	15
Стратегия ветвления	16
Верхняя оценка	17
Нижняя оценка	17
2.3. Задание	18
2.4. Тестовые задачи	18
Литература	27

Информационные технологии в области принятия решения. Часть 1

В учебно-методической разработке приводятся задания для лабораторных работ 1 и 2 спецсеминара «Информационные технологии в области принятия» магистров 1 курса направления подготовки «Прикладная информатика» 09.04.03, обучающихся по магистерской программе «Прикладная информатика в области принятия решений». По каждой из лабораторных работ даются постановки прикладных задач, описываются подходы к их решению, формулируется задание и приводятся тестовые задачи, необходимые для выполнения заданий.

1. Лабораторная работа «Задача одномерного раскроя»

Рассматривается задача одномерного раскроя. В качестве метода решения рассматриваются два эвристических подхода: жадный одношаговый алгоритм и итерационные стратегии, основанные на жадном алгоритме. Задание заключается в том, чтобы

- предложить собственную реализацию итерационной стратегии,
- провести вычислительный эксперимент с использованием тестовых задач,
- на основании вычислительного эксперимента обосновать целесообразность предложенной итерационной стратегии.

1.1. Математическая модель

Содержательная постановка

Имеются прутья равной длины и заготовки заданной длины, которые необходимо нарезать из прутьев. Требуется построить вариант раскроя прутьев на заготовки, требующий минимального числа прутьев.

Исходные параметры

Заданы L – длина прутьев, n – количество заготовок, l_i – длина заготовки номер i , $l_i \leq L$, $i = \overline{1, n}$. Будем считать, что все исходные параметры принимают натуральные значения.

Варьируемые параметры

Пусть y – количество использованных прутьев, x_i – номер прута, из которого будет изготовлена заготовка i , $i = \overline{1, n}$.

Ограничения математической модели

$$y \in N, \tag{1.1}$$

$$x_i \in \{1, 2, \dots, y\}, i = \overline{1, n}, \tag{1.2}$$

$$\sum_{i|x_i=j} l_i \leq L, j = \overline{1, y}. \tag{1.3}$$

Критерии оптимальности

$$y \rightarrow \min. \tag{1.4}$$

Анализ сложности

Задача одномерного раскроя ставится как задач (1.1)-(1.4). Данная задача относится к классу NP-трудных задач, т.к. к ней сводится за полиномиальное время NP-полная задача о разбиение [1].

1.2. Жадный алгоритм и итерационные стратегии

Жадный алгоритм

Пусть дана перестановка $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ множества номеров перестановок. Для заданной перестановки π можно предложить следующий жадный алгоритм раскроя прутьев.

Алгоритм 1.1. (жадный алгоритм)

Вход: $\pi = (\pi_1, \pi_2, \dots, \pi_n)$, L , n , l_i , $i = \overline{1, n}$.

Выход: y , x_i , $i = \overline{1, n}$.

Шаг 1. Пусть $y = 1$, $i = 1$, $x_{\pi_1} = 1$ (вырезаем заготовку π_1 из первого прута). Далее $i = i + 1$, перейти на шаг 2.

Шаг 2. Если $i > n$, то алгоритм завершен, иначе перейти на шаг 3.

Шаг 3. Определить номер z' первого из используемых на данный момент прутьев, из которого с учетом уже сделанных раскроев можно вырезать заготовку π_i :

$$z' = \min Z, \text{ где } Z = \{z \mid \sum_{k \mid x_{\pi_k} = z, k \in \{1, 2, \dots, i-1\}} l_k + l_{\pi_i} \leq L, z \in \{1, 2, \dots, y\}\}.$$

Если такой номер z' существует (т.е. если $Z \neq \emptyset$), то пусть $x_{\pi_i} = z'$, $i = i + 1$ (вырезаем заготовку π_i из имеющегося прута z'); иначе $y = y + 1$, $x_{\pi_i} = y$, $i = i + 1$ (вырезаем заготовку π_i из нового прута). Перейти на шаг 2.

Несложно увидеть, что алгоритм 1.1 находит допустимое решение задачи одномерного раскроя. Данный алгоритм стремится вырезать каждую последующую заготовку (при заданной перестановке заготовок) из имеющихся прутьев, если это не удастся, то заготовка вырезается из следующего прута. Такое поведение позволяет классифицировать алгоритм как «жадный» [2].

Итерационные стратегии

Предложенный жадный алгоритм строит для заданной перестановки некоторое допустимое решение задачи. При этом очевидно, что существует такая перестановка, для которой жадный алгоритм 1.1 построит допустимое решение, которое будет являться оптимальным решением задачи одномерного раскроя. С другой стороны общее количество перестановок равно $n!$ и таким образом полный перебор всех перестановок на практике невозможен.

Данный факт наталкивает на идею построения эвристического итерационного алгоритма, заключающегося в переборе ограниченного числа перестановок из некоторого множества «хороших» перестановок. Для каждой из таких перестановок выполняется жадный алгоритм 1.1 и фиксируется рекордное значение критерия задачи одномерного раскроя.

Алгоритм 1.2. (итерационная стратегия)

Вход: Множество перестановок Π , L , n , l_i , $i = \overline{1, n}$.

Выход: y^* , x_i^* , $i = \overline{1, n}$.

Шаг 1. Пусть $y^* = +\infty$. Выбрать перестановку $\pi \in \Pi$. Перейти на шаг 2.

Шаг 2. Выполнить жадный алгоритм 1.1 для перестановки π . Пусть y , x_i , $i = \overline{1, n}$ – решение, найденное алгоритмом 1.1. Если $y < y^*$, то $y^* = y$, $x_i = x_i^*$, $i = \overline{1, n}$. Перейти на шаг 3.

Шаг 3. Если обработаны все перестановки множества Π , то алгоритм завершен; иначе выбрать очередную перестановку $\pi \in \Pi$ и перейти на шаг 2.

Данная итерационная стратегия может быть классифицирована как эвристическая итерационная стратегия, т.к. основана на построении множества «хороших» перестановок. В качестве такого множества перестановок могут быть предложены следующие варианты.

Стратегия 1

Идея стратегии: первыми укладываем «большие заготовки», далее остатки прутьев заполняются «маленькими» заготовками.

Предложим следующую реализацию данной стратегии. Необходимо упорядочить все заготовки по убыванию их длины. Среди них взять первые $w\%$ заготовок и обработать в порядке следования, где w заданное число процентов. Далее оставшиеся заготовки упорядочить по возрастанию их длины и обработать их в данном порядке. Данная стратегия определяет следующее правило построения перестановки.

Не уменьшая общности, будем считать, что множество заготовок упорядочено по убыванию их длины: $l_1 \geq l_2 \geq \dots \geq l_n$. Пусть задан параметр $w \in [0, 100]$, тогда $k = \lfloor n \cdot w / 100 \rfloor$ и перестановка $\pi(w) = (\pi_1, \pi_2, \dots, \pi_n)$ определяется следующим образом:

$$\pi_1 = 1, \pi_2 = 2, \dots, \pi_k = k, \pi_{k+1} = n, \pi_{k+2} = n-1, \dots, \pi_n = k+1.$$

Далее пусть задано число шагов $N \geq 2$ итерационного алгоритма. Тогда определим множество перестановок следующим образом

$$\Pi = \{ \pi(w) \mid w = t \cdot 100 / (N-1), t = \overline{0, N-1} \}.$$

Стратегия 2

Идея стратегии: первыми укладываем «большие заготовки», далее остатки прутьев заполняются оставшимися заготовками в случайно сгенерированном порядке.

Предложим следующую реализацию данной стратегии. Необходимо упорядочить все заготовки по убыванию их длины. Среди них взять первые $w\%$ заготовок и обработать в порядке следования. Далее оставшиеся заготовки упорядочить случайным образом и обработать их в данном порядке. Данная стратегия определяет следующее правило построения перестановки.

Не уменьшая общности, будем считать, что множество заготовок упорядочено по убыванию их длины: $l_1 \geq l_2 \geq \dots \geq l_n$. Пусть задан параметр $w \in [0, 100]$, тогда $k = \lfloor n \cdot w / 100 \rfloor$. Далее пусть $\varphi_1, \varphi_2, \dots, \varphi_{n-k}$ – случайно

сгенерированная перестановка номеров заготовок $k+1, k+2, \dots, n$. Тогда перестановка $\pi(w) = (\pi_1, \pi_2, \dots, \pi_n)$ строится следующим образом:

$$\pi_1 = 1, \pi_2 = 2, \dots, \pi_k = k, \pi_{k+1} = \varphi_1, \pi_{k+2} = \varphi_2, \dots, \pi_n = \varphi_{n-k}.$$

Далее пусть заданы параметры $N_1 \geq 2$, $N_2 \geq 1$ итерационного алгоритма. Будем рассматривать перестановки вида $\pi(w)$, где $w = t \cdot 100 / (N_1 - 1)$, $t = \overline{0, N_1 - 1}$, и каждую из перестановок вида $\pi(w)$ будем генерировать N_2 раза. Таким образом строится $N_1 \cdot N_2$ перестановок (возможно с повторениями).

1.3. Задание

При выполнении лабораторной работы необходимо:

1. для каждой из задач построить нижнюю оценку $y^H = \left\lceil \sum_{i=1}^n l_i / L \right\rceil$,
2. реализовать жадный алгоритм, где в качестве перестановки рассматривается список прутьев, упорядоченный по убыванию (*базовый алгоритм*),
3. для каждой из тестовых задач оценить отклонение базового алгоритма от нижней оценки как $(y^o - y^H) / y^H$, где y^o – найденное значение критерия,
4. оценить среднее отклонение базового алгоритма от нижней оценки,
5. предложить свою реализацию жадного алгоритма, который дает (в среднем по тестовым задачам) лучшее отклонение от нижней оценки по сравнению с базовым алгоритмом,
6. предложить собственную реализацию стратегии для итерационного алгоритма,
7. построить график изменения отклонения от нижней оценки при увеличении числа шагов итерационного алгоритма.

Требования к программной реализации:

1. жадный алгоритм должен быть реализован в виде функции, принимающей в качестве входного параметра перестановку заготовок,
2. итерационная стратегия должна предоставлять возможность перебирать перестановки,
3. итерационный алгоритм должен, перебирая перестановки при помощи итерационной стратегии, запускать для каждой из них жадный алгоритм,
4. итерационный алгоритм должен принимать итерационную стратегию в качестве входного параметра,
5. программная система должна предоставлять возможность запускать выбранный набор тестовых задач, используя выбранный набор итерационных стратегий, и формировать на основании расчета таблицу отклонений каждой из задач на каждой и стратегий, в том числе средних отклонений по каждой стратегии,
6. программная система должна сохранять найденные рекордные решения тестовых задач.

1.4. Тестовые задачи

Формат описания исходных данных: приводятся длина прутьев и число заготовок, далее через символ пробела приводятся длины всех заготовок.

Задача 1.

$L = 18, n = 10$, длины заготовок: 10 8 7 8 3 5 12 5 8 4

Известный рекорд: $y^o = 4$.

Прут 1: 5 8 5

Прут 2: 10 8

Прут 3: 12 4

Прут 4: 7 8 3

Задача 2.

$L = 18, n = 10$, длины заготовок: 5 8 5 5 11 5 7 6 7 11

Известный рекорд: $y^o = 4$.

Прут 1: 11 7

Прут 2: 11 7

Прут 3: 8 5 5

Прут 4: 5 5 6

Задача 3.

$L = 26, n = 100$, длины заготовок: 14 4 4 12 5 5 10 6 6 14 4 4 7 7 7 10 6 6 8
6 6 15 4 4 9 6 6 10 6 6 11 5 5 12 5 5 13 5 5 8 6 6 10 15 1 10 6 6 14 4 4
8 16 1 1 11 5 5 12 5 5 14 4 4 16 4 4 12 5 5 8 7 7 8 6 6 15 4 4 8 6 6 16
4 4 8 6 6 10 6 6 14 4 4 16 4 4 11 5 5

Известный рекорд: $y^o = 29$.

Прут 1: 16 10

Прут 2: 16 10

Прут 3: 16 10

Прут 4: 16 10

Прут 5: 15 11

Прут 6: 15 11

Прут 7: 15 11

Прут 8: 14 12

Прут 9: 14 12

Прут 10: 14 12

Прут 11: 14 12

Прут 12: 14 10 1 1

Прут 13: 13 10 1

Прут 14: 9 8 8

Прут 15: 8 8 8

Прут 16: 8 8 7

Прут 17: 7 7 4 4 4

Прут 18: 4 4 4 4 4 4

Прут 19: 4 4 4 4 4 4

Прут 20: 4 4 4 4 4 5

Прут 21: 5 5 5 5 5

Прут 22: 5 5 5 5 5

Прут 23: 5 5 5 5 5

Прут 24: 6 6 6 6

Прут 25: 6 6 6 6

Прут 26: 6 6 6 6

Прут 27: 6 6 6 6

Прут 28: 6 6 6 6

Прут 29: 6 6 7 7

Задача 4.

$L = 27, n = 101$, длины заготовок: 16 4 4 15 4 4 11 6 6 9 7 7 10 6 6 15 8 1 1
13 5 5 12 13 1 1 8 7 7 16 4 4 9 6 6 9 6 6 10 6 6 12 6 6 10 6 6 15 4 4 10
6 6 13 5 5 14 5 5 15 4 4 15 4 4 15 4 4 13 5 5 15 4 4 15 4 4 10 6 6 9 6 6
16 4 4 9 6 6 11 6 6 12 5 5 12 5 5 9 7 7

Известный рекорд: $y^o = 29$.

Прут 1: 7 6 6 8

Прут 2: 16 9 1 1

Прут 3: 13 4 10

Прут 4: 6 6 15

Прут 5: 6 15 1 4 1

Прут 6: 16 6 5

Прут 7: 16 7 4
 Прут 8: 15 6 6
 Прут 9: 9 15
 Прут 10: 11 6 6 4
 Прут 11: 6 6 6 9
 Прут 12: 10 6 5 5
 Прут 13: 5 6 11 5
 Прут 14: 6 6 14
 Прут 15: 6 6 4 9
 Прут 16: 13 6 6
 Прут 17: 15 6 6
 Прут 18: 4 7 7 5 4

Прут 19: 13 5 5 4
 Прут 20: 15 6 5
 Прут 21: 5 5 5 4 4 4
 Прут 22: 15 4 4 4
 Прут 23: 15 4 7
 Прут 24: 4 4 4 4 4 7
 Прут 25: 13 12
 Прут 26: 12 12
 Прут 27: 10 12
 Прут 28: 10 10
 Прут 29: 9 9 8

Задача 5.

$L = 23, n = 101$, длины заготовок: 23 12 4 4 7 6 6 8 5 5 7 6 6 7 6 6 7 6 6 9 5
 5 10 5 5 12 4 4 7 6 6 9 5 5 10 5 5 11 4 4 10 5 5 9 5 5 13 4 4 13 4 4 13
 4 4 8 13 1 1 8 5 5 11 4 4 11 4 4 9 5 5 11 4 4 9 5 5 11 4 4 8 5 5 10 10 1
 1 8 5 5 13 4 4 11 4 4 11 4 4 9 5 5

Известный рекорд: $y^o = 29$.

Задача 6.

$L = 296, n = 1002$, длины заготовок: 180 43 43 183 40 40 179 43 43 124 62 62
 167 47 47 139 58 58 179 42 42 92 74 74 163 49 49 186 88 144 55 55 156 52
 52 183 40 40 98 74 74 104 70 70 154 107 117 117 167 120 139 58 58 104 68
 68 108 70 70 168 47 47 152 106 166 47 47 101 72 72 185 41 41 104 140 141
 57 57 170 46 46 172 90 171 46 46 95 174 170 170 177 42 42 159 112 140 57
 57 117 154 106 106 187 38 38 155 109 185 39 39 168 168 92 150 178 178
 166 46 46 119 153 179 43 43 109 109 115 147 177 44 44 170 45 45 181 181
 144 54 54 105 68 68 154 118 129 61 61 138 57 57 175 175 134 60 60 89 73
 73 158 130 178 42 42 100 72 72 101 72 72 141 141 168 168 159 51 51 166
 48 48 114 140 186 186 168 168 164 164 103 169 131 61 61 118 162 119 176
 135 160 172 44 44 127 59 59 106 148 95 75 75 155 50 50 119 66 66 180 43
 43 183 41 41 126 62 62 103 71 71 153 54 54 114 114 119 66 66 89 89 184
 89 124 63 63 150 53 53 89 73 73 142 54 54 176 110 89 74 74 91 74 74 129
 164 129 62 62 158 51 51 166 46 46 152 53 53 150 53 53 147 53 53 146 55
 55 108 67 67 177 107 168 103 174 118 169 47 47 161 48 48 106 71 71 187
 40 40 175 44 44 182 40 40 121 153 119 63 63 136 143 111 162 160 49 49
 182 100 151 110 172 45 45 165 46 46 160 160 131 131 173 44 44 142 142
 118 67 67 135 58 58 150 137 116 65 65 107 165 183 41 41 171 45 45 96 70
 70 115 67 67 109 109 100 100 128 60 60 152 53 53 104 71 71 111 68 68 137
 139 141 141 173 173 142 131 94 72 72 185 85 138 57 57 170 45 45 175 45
 45 115 65 65 103 144 113 149 93 93 158 51 51 123 152 99 149 145 54 54
 148 109 89 185 94 71 71 122 64 64 146 54 54 186 41 41 163 131 177 43 43
 187 187 174 174 100 140 122 172 163 163 173 173 130 59 59 164 164 124 64
 64 163 163 106 106 144 56 56 146 55 55 134 115 164 49 49 103 103 118 67
 67 151 151 109 109 110 67 67 140 140 173 173 158 158 117 67 67 187 40 40
 140 57 57 93 71 71 100 71 71 122 65 65 95 75 75 101 101 184 90 185 39 39
 111 148 100 165 96 96 103 71 71 164 46 46 185 39 39 99 73 73 107 70 70
 95 72 72 105 71 71 89 89 157 157 108 67 67 128 129 107 67 67 134 58 58
 141 141 94 94 137 123 113 68 68 156 109 158 49 49 141 141 102 72 72 150
 54 54 165 165 105 165 126 123 150 54 54 93 93 126 60 60 170 46 46 121 62
 62 103 70 70 176 89 120 64 64 177 43 43 126 63 63 124 61 61 183 40 40
 150 129 107 107 158 158 165 46 46 106 106 139 156 119 64 64 167 48 48 99
 161 166 92 127 62 62 135 57 57 183 41 41 183 42 42 129 62 62 92 173 114

155 164 164 95 71 71 173 45 45 173 44 44 104 70 70 118 65 65 168 168 143
56 56 124 153 126 60 60 95 95 178 42 42 139 57 57 153 51 51 108 66 66
140 56 56 152 53 53 148 53 53 155 50 50 98 98 156 156 96 72 72 117 66 66
119 126 105 105 174 174 166 48 48 181 40 40 150 55 55 153 120 167 46 46
117 148 130 60 60 155 52 52 119 65 65 130 61 61 143 56 56 106 67 67 129
61 61 102 102 137 137 101 72 72 153 153 96 71 71 163 47 47 146 146 104
104 151 114 108 68 68 99 99 159 49 49 103 160 159 50 50 91 73 73 111 69
69 164 49 49 126 60 60 149 53 53 180 43 43 173 118 116 63 63 127 127 92
76 76 158 99 119 64 64 141 146 104 165 173 46 46 119 119 98 98 163 48 48
155 117 169 47 47 117 145 173 102 102 72 72 159 159 93 148 147 56 56 123
123 178 43 43 111 111 90 90 158 49 49 107 167 167 47 47 183 83 142 55 55
129 60 60 179 112 102 141 182 182 169 93 146 54 54 171 115 180 42 42 170
45 45 111 68 68 165 124 144 54 54 103 68 68 92 74 74 119 119 112 173 182
43 43 123 62 62 150 52 52 184 99 122 64 64 149 53 53 102 69 69 105 69 69
169 47 47 159 159 150 52 52 89 74 74 152 51 51 118 63 63 170 45 45 103
103 163 49 49 183 183 161 161 142 57 57 171 45 45 113 68 68 147 54 54

Известный рекорд: $y^o = 347$.

Задача 7.

$L = 216$, $n = 1001$, длины заготовок: 132 30 30 98 98 72 129 134 29 29 104 40
40 134 30 30 135 29 29 129 129 133 30 30 107 40 40 118 34 34 91 44 44
113 38 38 80 50 50 123 34 34 79 125 118 37 37 131 31 31 110 110 84 47 47
116 116 107 38 38 129 31 31 135 29 29 117 37 37 91 44 44 85 47 47 118 36
36 125 86 123 34 34 117 37 37 131 30 30 76 100 119 34 34 129 73 79 118
68 55 55 105 39 39 92 46 46 89 47 47 83 83 76 50 50 69 54 54 73 110 118
118 103 40 40 126 71 110 110 84 47 47 117 117 94 105 121 121 123 123 88
47 47 130 130 107 40 40 66 53 53 67 67 102 41 41 71 54 54 70 70 108 86
122 122 70 54 54 85 48 48 93 45 45 66 123 114 38 38 66 66 126 32 32 108
38 38 93 43 43 74 53 53 116 36 36 72 72 101 43 43 117 117 101 42 42 123
34 34 116 89 78 78 103 103 120 35 35 96 43 43 125 125 106 40 40 135 29
29 84 49 49 76 76 88 45 45 113 113 123 85 116 116 101 43 43 78 50 50 108
38 38 80 51 51 130 86 121 88 83 48 48 120 120 65 54 54 118 77 109 39 39
74 103 71 54 54 72 130 119 36 36 92 46 46 71 52 52 136 136 95 45 45 118
36 36 135 135 127 32 32 96 42 42 119 94 128 128 66 54 54 120 36 36 71 54
54 100 41 41 130 31 31 75 109 87 48 48 123 123 134 64 89 45 45 113 80
121 36 36 105 95 70 53 53 84 48 48 80 50 50 123 35 35 130 31 31 87 87 79
50 50 130 31 31 120 34 34 104 40 40 106 41 41 76 50 50 104 104 109 39 39
118 36 36 103 103 89 45 45 113 37 37 119 119 102 42 42 74 53 53 74 134
86 86 74 52 52 104 39 39 77 51 51 65 55 55 122 122 112 83 83 49 49 91
119 111 38 38 94 45 45 98 44 44 130 31 31 118 35 35 121 34 34 115 115 65
56 56 125 125 136 76 117 117 95 43 43 104 104 121 84 114 76 99 43 43 79
51 51 96 96 99 99 106 41 41 122 91 114 37 37 118 36 36 128 33 33 77 50
50 95 88 108 76 103 103 72 52 52 70 54 54 82 129 136 29 29 85 103 80 48
48 100 43 43 86 47 47 116 116 113 36 36 134 30 30 83 124 68 68 93 46 46
104 100 133 30 30 73 52 52 126 33 33 101 85 122 35 35 91 44 44 106 107
78 50 50 96 96 81 81 123 82 72 72 113 38 38 82 101 75 75 106 106 82 49
49 134 29 29 66 56 56 135 75 119 119 136 29 29 85 48 48 133 77 71 71 68
68 73 53 53 95 96 78 49 49 115 115 103 106 69 52 52 71 126 119 35 35 122
80 136 136 87 47 47 124 33 33 130 32 32 105 105 96 44 44 130 66 118 36
36 129 129 109 109 80 48 48 126 32 32 105 40 40 74 74 95 95 89 119 128
83 105 105 100 88 89 46 46 72 110 76 76 66 55 55 73 102 131 85 117 117
96 43 43 130 82 130 62 93 93 65 55 55 69 52 52 100 42 42 73 101 94 45 45
121 91 129 76 84 48 48 118 80 119 36 36 84 84 117 89 129 32 32 102 92
103 103 136 29 29 69 52 52 67 67 70 70 97 44 44 90 101 111 39 39 77 52

52 127 32 32 121 72 132 31 31 107 107 69 52 52 87 48 48 110 39 39 78 108
 99 101 109 40 40 83 48 48 103 103 107 107 109 38 38 126 126 120 120 126
 33 33 135 57 126 32 32 133 75 131 131 134 65 106 40 40 117 90 103 41 41
 78 78 79 49 49 110 39 39 94 94 81 50 50 103 103 101 41 41 111 39 39 72
 53 53 136 30 30 89 47 47 118 118 119 36 36 113 38 38 106 39 39 74 53 53
 113 38 38 126 32 32 99 44 44 116 73 103 41 41 71 105 90 45 45 72 113 116
 37 37 119 80 84 48 48 118 118 84 49 49 131 31 31 97 42 42 110 37 37 73
 52 52 90 90 120 120 108 40 40 108 108 69 54 54 97 43 43 84 49 49 86 86
 114 76 75 75 125 85 110 82 125 33 33 68 68 119 34 34 104 41 41 119 97
 112 112 120 34 34 77 52 52 86 47 47 65 56 56 119 34 34 135 135 125 33 33
 72 52 52 102 102 112 38 38 128 31 31 77 52 52 71 71 85 49 49 66 66 104
 40 40 73 51 51 125 125 123 33 33 106 41 41

Известный рекорд: $y^o = 351$.

Задача 8.

$L = 219$, $n = 1001$, длины заготовок: 105 42 42 79 51 51 105 42 42 72 126 95 92
 83 83 130 33 33 69 69 120 35 35 112 39 39 77 50 50 70 53 53 93 47 47 87
 48 48 81 49 49 74 74 108 108 130 130 89 116 94 44 44 101 43 43 127 88
 125 34 34 67 146 78 51 51 87 47 47 75 53 53 134 31 31 120 37 37 112 95
 97 44 44 72 106 99 44 44 111 40 40 116 116 80 49 49 89 46 46 78 50 50
 132 32 32 134 134 73 51 51 78 52 52 108 108 107 107 114 39 39 84 49 49
 70 53 53 105 40 40 96 43 43 126 33 33 95 45 45 90 90 95 44 44 106 41 41
 129 32 32 78 127 81 81 132 32 32 128 33 33 94 91 71 52 52 117 37 37 118
 36 36 90 47 47 132 33 33 126 33 33 68 68 126 34 34 123 35 35 104 42 42
 130 130 118 36 36 68 109 118 35 35 90 111 89 120 138 138 124 124 101 96
 112 39 39 128 32 32 129 65 116 37 37 107 101 105 105 121 37 37 138 78
 134 32 32 132 32 32 127 33 33 74 54 54 138 28 28 116 38 38 74 52 52 138
 138 92 102 100 42 42 136 136 92 104 113 38 38 76 52 52 81 49 49 79 52 52
 114 114 135 135 128 34 34 105 43 43 97 43 43 79 79 69 54 54 72 109 110
 40 40 79 50 50 116 116 71 141 77 77 114 39 39 78 78 118 36 36 123 36 36
 66 66 77 50 50 135 30 30 80 49 49 87 49 49 90 90 78 51 51 73 52 52 138
 30 30 82 82 89 47 47 126 68 116 116 94 94 87 97 76 53 53 83 48 48 104 42
 42 115 38 38 86 48 48 114 114 111 40 40 131 32 32 116 80 80 51 51 134 82
 108 42 42 127 34 34 132 32 32 121 36 36 80 80 104 41 41 103 42 42 106 87
 121 121 84 48 48 130 79 125 34 34 82 123 103 42 42 117 117 82 82 126 126
 101 108 115 37 37 116 37 37 110 110 79 104 96 45 45 121 121 118 94 124
 35 35 125 35 35 120 36 36 71 105 121 36 36 80 50 50 75 51 51 110 39 39
 117 117 136 136 102 44 44 73 73 118 118 70 125 135 77 134 63 84 49 49
 138 69 72 54 54 115 37 37 136 29 29 71 130 128 32 32 126 67 90 46 46 82
 49 49 107 107 119 35 35 133 32 32 85 85 93 46 46 93 93 130 130 93 47 47
 134 134 71 52 52 96 99 112 38 38 128 33 33 73 53 53 94 44 44 87 105 126
 89 90 48 48 98 42 42 130 32 32 98 98 74 54 54 75 52 52 136 72 65 65 85
 48 48 117 117 78 49 49 124 35 35 77 77 81 49 49 125 125 136 31 31 114 38
 38 125 33 33 127 33 33 131 33 33 76 51 51 67 56 56 117 117 90 48 48 94
 46 46 132 132 76 76 127 33 33 83 48 48 72 133 75 52 52 98 43 43 88 127
 66 66 67 54 54 110 38 38 103 96 123 123 121 35 35 68 129 131 31 31 133
 66 107 42 42 79 49 49 72 52 52 70 110 111 40 40 112 39 39 76 53 53 67
 152 84 49 49 125 125 79 109 74 74 116 38 38 126 126 134 30 30 72 53 53
 71 54 54 121 35 35 94 45 45 89 47 47 131 67 97 45 45 126 126 119 89 108
 41 41 120 35 35 75 142 85 48 48 131 131 69 53 53 102 43 43 116 116 137
 80 125 33 33 92 46 46 119 119 123 78 89 48 48 107 41 41 110 110 93 45 45
 93 47 47 121 35 35 88 109 97 97 127 84 71 54 54 120 35 35 72 72 116 38
 38 124 78 79 52 52 96 96 104 104 126 34 34 127 34 34 129 34 34 80 51 51
 128 32 32 133 31 31 111 111 85 48 48 70 121 134 32 32 103 43 43 91 46 46

138 138 131 32 32 81 51 51 83 83 65 65 76 76 111 40 40 104 43 43 89 89
 96 91 125 125 112 39 39 127 33 33 138 138 119 36 36 86 120 119 119 130
 72 123 34 34 134 63 67 57 57 137 31 31 81 49 49 135 31 31 89 48 48 77 77
 130 130 108 40 40 79 52 52 104 104 122 122 104 42 42 113 38 38 106 106
 70 127 68 56 56 117 37 37 106 41 41 112 38 38 123 36 36 115 37 37 78 52
 52 97 44 44 107 82 110 110 80 80 67 137 105 41 41 100 43 43 137 29 29 94
 94 68 54 54 126 33 33 119 92 114 37 37 90 90 97 46 46 78 51 51 101 42 42
 119 36 36 71 71 125 33 33 83 83

Известный рекорд: $y^o = 347$.

Задача 9.

$L = 263$, $n = 1000$, длины заготовок: 153 82 128 50 50 84 84 150 42 42 92 63
 63 125 134 142 104 97 59 59 78 68 68 89 148 113 112 124 50 50 113 56 56
 105 59 59 158 38 38 129 97 114 52 52 123 123 105 131 98 58 58 149 42 42
 94 61 61 143 44 44 152 39 39 113 124 132 47 47 159 87 156 39 39 132 49
 49 145 104 89 89 142 43 43 146 43 43 101 59 59 151 40 40 84 84 138 46 46
 138 44 44 104 146 159 103 128 49 49 164 71 142 44 44 87 158 164 71 134
 47 47 136 47 47 119 52 52 87 87 101 151 157 38 38 162 36 36 79 67 67 160
 160 123 52 52 129 47 47 141 43 43 137 44 44 102 58 58 88 62 62 156 40 40
 142 44 44 125 51 51 103 56 56 155 40 40 119 139 142 142 142 142 118 54
 54 87 64 64 159 37 37 87 64 64 143 143 114 53 53 98 98 162 36 36 160 160
 148 97 149 41 41 112 54 54 148 40 40 148 41 41 86 62 62 155 39 39 102
 136 83 66 66 122 122 146 41 41 147 93 109 114 104 146 91 91 161 82 164
 70 163 163 101 58 58 121 50 50 164 35 35 147 91 137 137 101 60 60 88 128
 135 48 48 147 41 41 161 38 38 99 59 59 91 62 62 119 52 52 148 40 40 111
 111 97 161 123 126 83 83 142 142 154 39 39 138 44 44 115 115 114 114 157
 87 88 88 124 52 52 121 52 52 85 65 65 137 108 131 131 160 36 36 84 64 64
 81 81 147 42 42 111 111 132 47 47 98 61 61 114 55 55 89 63 63 129 48 48
 155 155 140 46 46 86 64 64 145 86 152 152 122 122 100 100 162 37 37 160
 37 37 123 50 50 136 136 126 50 50 144 44 44 91 64 64 142 43 43 90 148
 103 60 60 108 148 155 39 39 91 62 62 126 49 49 154 39 39 113 115 103 57
 57 106 56 56 101 58 58 112 56 56 151 151 90 64 64 146 43 43 87 87 131
 131 114 53 53 147 113 132 46 46 104 104 82 63 63 125 50 50 166 166 146
 97 141 43 43 86 65 65 92 92 89 89 142 142 158 37 37 120 52 52 146 43 43
 125 125 86 86 150 41 41 83 83 92 130 166 92 141 43 43 161 36 36 94 135
 116 116 147 41 41 115 52 52 112 54 54 99 160 164 36 36 88 88 161 102 159
 80 145 97 151 151 145 42 42 146 43 43 118 118 158 38 38 151 91 142 45 45
 154 154 123 123 158 158 164 36 36 96 136 112 53 53 94 63 63 151 86 92 64
 64 117 117 107 55 55 94 60 60 133 133 161 37 37 162 77 101 58 58 123 51
 51 124 124 121 51 51 121 125 125 51 51 124 51 51 152 40 40 102 60 60 144
 44 44 154 90 81 67 67 153 103 120 52 52 100 135 142 142 99 99 122 51 51
 148 41 41 108 54 54 82 165 161 161 115 55 55 82 64 64 153 41 41 87 87
 157 157 130 130 126 50 50 140 103 118 118 87 140 90 163 158 39 39 122
 108 134 47 47 166 166 150 40 40 144 42 42 144 42 42 122 50 50 100 61 61
 95 63 63 126 126 152 152 95 59 59 125 50 50 104 157 129 47 47 85 85 106
 58 58 127 50 50 105 58 58 127 127 153 153 157 37 37 84 67 67 102 102 120
 52 52 166 166 148 41 41 126 49 49 158 158 115 55 55 111 54 54 147 42 42
 126 51 51 132 46 46 154 38 38 84 126 152 40 40 127 51 51 137 44 44 150
 111 163 36 36 108 55 55 156 39 39 153 39 39 86 64 64 123 49 49 140 43 43
 83 67 67 123 105 90 90 103 59 59 117 131 158 38 38 151 85 106 59 59 128
 128 120 52 52 93 123 89 65 65 159 90 151 112 145 42 42 107 107 146 146
 157 39 39 161 38 38 83 66 66 131 48 48 108 56 56 162 36 36 132 46 46 142
 142 90 65 65 160 38 38 93 62 62 101 60 60 85 66 66 154 40 40 83 64 64 94
 61 61 122 122 115 55 55 117 117 164 164 120 51 51 101 145 152 41 41 152

42 42 140 45 45 133 46 46 89 89 114 56 56 109 56 56 97 60 60 148 148 101
 59 59 131 48 48 131 49 49 97 135 108 108 127 50 50 147 147 102 102 156
 100 145 44 44 152 40 40 105 105 90 65 65 130 130 156 156 166 93 89 150
 108 55 55 154 40 40 144 144 124 113 122 135 154 40 40 81 68 68 148 42 42
 94 154 150 40 40 103 59 59 80 66 66 92 64 64 147 101 156 40 40 131 131

Известный рекорд: $y^o = 348$.

Задача 10.

$L = 249$, $n = 1000$, длины заготовок: 80 140 145 38 38 98 55 55 93 55 55 119
 48 48 142 40 40 137 40 40 92 92 122 118 126 46 46 135 43 43 121 45 45 82
 60 60 98 141 101 118 83 58 58 139 39 39 86 60 60 111 51 51 137 80 116
 116 142 40 40 140 39 39 142 39 39 91 58 58 151 95 103 53 53 98 56 56 154
 35 35 145 37 37 111 111 102 53 53 139 41 41 147 147 117 46 46 92 92 154
 154 141 141 138 40 40 117 117 105 53 53 95 146 90 119 102 53 53 92 59 59
 141 141 75 75 96 57 57 93 57 57 74 65 65 145 38 38 77 62 62 108 53 53
 114 114 79 149 118 49 49 140 40 40 146 146 122 46 46 95 56 56 118 48 48
 145 37 37 81 62 62 102 55 55 75 64 64 134 113 104 52 52 117 49 49 95 95
 152 36 36 88 58 58 122 93 133 90 145 38 38 147 147 117 117 146 37 37 106
 125 113 100 101 101 125 45 45 90 57 57 134 42 42 99 53 53 91 91 145 37
 37 144 144 106 51 51 140 39 39 106 50 50 124 124 118 48 48 82 82 86 60
 60 79 79 87 60 60 139 40 40 104 51 51 148 35 35 152 83 112 105 136 136
 86 61 61 118 49 49 130 130 74 61 61 109 49 49 93 93 102 102 142 142 76
 125 134 42 42 119 47 47 127 44 44 151 97 116 48 48 75 64 64 95 55 55 146
 81 145 38 38 84 135 133 133 86 58 58 135 41 41 82 60 60 116 48 48 141
 141 135 40 40 75 75 99 55 55 139 105 93 58 58 85 59 59 79 63 63 140 87
 120 47 47 127 45 45 141 39 39 112 51 51 152 152 150 36 36 98 56 56 105
 104 123 45 45 144 38 38 98 122 140 39 39 97 56 56 78 170 121 47 47 156
 71 111 51 51 140 39 39 86 146 130 99 102 102 123 45 45 122 45 45 134 42
 42 134 41 41 125 117 121 47 47 102 53 53 144 38 38 93 140 98 133 133 43
 43 136 42 42 88 88 126 45 45 145 36 36 78 63 63 101 101 81 60 60 80 80
 129 45 45 132 42 42 118 48 48 77 77 116 49 49 110 110 107 107 155 35 35
 120 120 137 41 41 93 55 55 138 40 40 88 57 57 128 44 44 144 37 37 78 78
 132 132 119 126 149 149 131 42 42 111 49 49 127 43 43 118 118 90 59 59
 143 38 38 153 35 35 93 121 134 42 42 110 51 51 150 80 111 111 81 124 134
 134 154 154 119 47 47 82 157 87 135 91 115 84 154 79 64 64 135 41 41 117
 48 48 141 38 38 148 85 125 46 46 139 40 40 151 35 35 149 36 36 98 57 57
 96 54 54 100 128 134 41 41 139 139 150 35 35 94 55 55 153 85 88 58 58 91
 91 84 143 143 39 39 93 57 57 120 45 45 122 122 154 76 121 121 129 43 43
 102 53 53 75 62 62 85 116 157 71 136 40 40 125 91 140 38 38 136 97 131
 84 132 106 102 53 53 149 36 36 116 47 47 150 37 37 119 46 46 105 52 52
 133 43 43 149 85 86 59 59 142 39 39 99 56 56 101 53 53 147 99 145 145
 111 111 148 79 91 57 57 100 54 54 156 156 155 35 35 146 36 36 151 35 35
 103 51 51 145 38 38 132 42 42 120 115 97 151 129 99 75 61 61 87 59 59
 145 37 37 134 134 82 164 109 51 51 97 55 55 155 35 35 141 141 77 77 116
 48 48 133 133 146 38 38 98 98 91 139 75 150 155 34 34 134 88 150 150 133
 99 109 51 51 92 92 131 42 42 128 42 42 142 38 38 74 74 98 54 54 88 57 57
 81 60 60 132 42 42 99 54 54 75 63 63 94 57 57 145 145 91 58 58 93 155
 131 131 120 94 126 44 44 135 135 100 56 56 149 36 36 83 83 145 39 39 116
 131 113 49 49 103 103 147 38 38 94 57 57 119 46 46 93 93 127 44 44 96
 118 129 112 143 143 125 45 45 154 154 99 54 54 119 119 105 105 145 38 38
 80 80 97 135 143 96 131 44 44 123 123 94 131 123 46 46 84 61 61 142 38
 38 87 147 155 34 34 134 98 88 59 59 151 35 35 140 40 40 129 44 44 125 46
 46 133 43 43 87 60 60 104 54 54 124 44 44 85 59 59 137 41 41 116 116 104
 131 86 58 58 142 80 93 57 57 124 124 97 56 56 122 46 46 141 141 145 38

38 103 55 55 110 99 76 76 86 61 61 84 61 61 152 152 116 116 124 124 150
35 35 110 139 127 127 152 34 34

Известный рекорд: $y^o = 346$.

2. Лабораторная работа «Задача о доставке заказов»

Рассматривается задача доставки заказов. В качестве метода решения рассматриваются метод ветвей и границ. Задание заключается в том, чтобы

- предложить реализацию метода ветвей и границ: предложить варианты основных стратегий метода, а именно ветвления, построения верхней и нижней оценок;
- провести вычислительный эксперимент с использованием тестовых задач,
- на основании вычислительного эксперимента обосновать целесообразность выбора предложенной комбинации стратегий метода ветвей и границ.

2.1. Математическая модель

Содержательная постановка

В логистическом центре имеются заказы, каждый из которых необходимо доставить в свой пункт назначения. По каждому из заказов задан директивный срок – время, не позже которого заказа должен быть доставлен. Задано время, требуемое для перемещения из логистического центра в пункты назначения, и время, требуемое для перемещения между пунктами назначения. Требуется определить такой порядок доставки заказов, при котором число заказов с нарушенным директивным сроком будет минимальным.

Исходные параметры

Заданы n – количество заказов; t_i^D – директивный срок заказа i , $i = \overline{1, n}$; t_{ij} – время на перемещение между пунктами назначения заказов i и j , $i, j = \overline{1, n}$; t_{0i} – время на перемещение между логистическим центром и пунктом назначения i , $i = \overline{1, n}$.

Будем считать, что все временные параметры принимают целые неотрицательные значения. Также будем предполагать, что времена на перемещение между пунктами удовлетворяют неравенству треугольника:

$$t_{ij} \leq t_{ik} + t_{jk}, \quad i, j, k = \overline{0, n}. \quad (2.1)$$

Содержательно данное условие означает, что время на перемещение из пункта i в пункт j не может быть строго большим, чем время на перемещение из пункта i в пункт j через промежуточный пункт k .

Варьируемые параметры

Пусть x_i – номер заказа, выполненный i -ым, $i = \overline{1, n}$.

Ограничения математической модели

$$x_i \in \{1, 2, \dots, n\}, \quad i = \overline{1, n}, \quad (2.1)$$

$$x_i \neq x_j, i \neq j, i, j = \overline{1, n}. \quad (2.2)$$

Критерии оптимальности

Обозначим через $y(i, x)$ номер, которым будет выполнен заказ i , если заказы выполняются в порядке определенном вектором $x = (x_1, x_2, \dots, x_n)$, т.е. $x_{y(i, x)} = i$, $i = \overline{1, n}$. Тогда обозначим через $z(i, x) = t_{0x_i} + \sum_{j=1}^{y(i, x)-1} t_{x_j x_{j+1}}$ момент времени, в который заказа i поступит в свой пункт назначения, если заказы выполняются в порядке определенном вектором x , $i = \overline{1, n}$. Наконец обозначим через

$$w(i, x) = \begin{cases} 0, & \text{если } z(i, x) \leq t_i^D \\ 1, & \text{иначе} \end{cases}$$

величину равную 0, если заказ i доставлен без нарушения директивного срока, и 1, иначе, $i = \overline{1, n}$.

Используя введенные обозначения, определим следующий критерий

$$\sum_{i=1}^n w(i, x) \rightarrow \min. \quad (2.3)$$

Анализ сложности

Задача доставки заказов ставится как задач (2.1)-(2.3). Данная задача относится к классу NP-трудных задач, т.к. к ней сводится за полиномиальное время NP-полная задача проверки гамильтоновости графа [1].

2.2. Метод ветвей и границ и стратегии ветвления, построения нижней и верхней оценок

Метод ветвей и границ

При реализации метода ветвей и границ (см. [2]) будем проводить ветвление по переменным. Каждую из вершин дерева ветвления будем представлять в виде $v = (\alpha_1, \alpha_2, \dots, \alpha_k)$, где k – уровень вершины (т.е. длина пути в дереве ветвлений от корня до данной вершины), $\alpha_1, \alpha_2, \dots, \alpha_k$ – зафиксированные значения первых k переменных вектора неизвестных x (т.е. $x_1 = \alpha_1, x_2 = \alpha_2, \dots, x_k = \alpha_k$). Замечание: если $k = 0$, то вершину будем записывать в виде $v = ()$. Через $B(v)$ будем обозначать верхнюю оценку вершины v , через $H(v)$ – нижнюю оценку вершины v , которые определяются следующим образом. С каждой из вершин $v = (\alpha_1, \alpha_2, \dots, \alpha_k)$ мы будем связывать оптимизационную задачу поиска вектора неизвестных x , удовлетворяющего системы ограничений (2.1), (2.2), (2.4) такого, что принимает оптимальное значений критерий (2.3).

$$x_i = \alpha_i, i = \overline{1, k}. \quad (2.4)$$

Пусть W^* – оптимальное значение критерия задачи (2.1)-(2.4). Тогда под верхней оценкой $B(v)$ понимается значение критерия задачи на некотором допустимом решении задачи (2.1)-(2.4), очевидно $W^* \leq B(v)$. Допустимое решение задачи (2.1)-(2.4), которое определяет верхнюю оценку $B(v)$ будем обозначать $x(B(v))$. С точки

зрения значения критерия верхняя оценка является достижимой. Под нижней оценкой $H(v)$ понимается такая величина, что $H(v) \leq W^*$. С точки зрения значения критерия нижняя оценка может являться недостижимой.

Метод ветвей и границ имеет следующую структуру:

Алгоритм 2.1. (метод ветвей и границ)

Вход: $n, t_i^D, i = \overline{1, n}; t_{ij}, i, j = \overline{0, n}$.

Выход: $x = (x_1, x_2, \dots, x_n)$.

Шаг 1. $V = \{()\}$ – инициализируем корень дерева ветвлений (на каждом из следующих шагов V будем являться множеством листьев дерева). Переход на шаг 2.

Шаг 2. (условие останова) Если $|V|=1$ и $B(v) = H(v)$, где $v \in V, x = x(B(v))$ и алгоритм завершен; иначе перейти на шаг 3.

Шаг 3. (шаг ветвления) Определить $v = \text{Ветвление}(V)$, где метод $\text{Ветвление}(V)$ возвращает некоторую вершину множества V . Пусть $v = (\alpha_1, \alpha_2, \dots, \alpha_k)$. Тогда обозначим $\{\beta_1, \beta_2, \dots, \beta_{n-k}\} = \{1, 2, \dots, n\} \setminus \{\alpha_1, \alpha_2, \dots, \alpha_k\}$. Далее

$$V = V \setminus \{v\} \cup \{(\alpha_1, \alpha_2, \dots, \alpha_k, \beta_i) \mid i = \overline{1, n-k}\}$$

(т.е. исключаем вершину v и добавляем всех ее потомков). Перейти на шаг 4.

Шаг 4. (отсечение) Для каждой пары вершин $v', v'' \in V, v' \neq v''$ выполнить: если $B(v') \leq H(v'')$, то $V = V \setminus \{v''\}$ (исключаем неперспективные направления). Перейти на шаг 2.

Реализация описанного метода ветвей и границ связана с 3 методами:

- $\text{Ветвление}(V)$, определяет стратегию ветвления,
- $B(v)$, определяет алгоритм нахождения верхней оценки,
- $H(v)$, определяет алгоритм нахождения нижней оценки.

Далее рассмотрим возможные подходы реализации данных методов.

Стратегия ветвления

В качестве стратегий ветвления могут быть рассмотрены следующие стратегии.

Обход в ширину. $\text{Ветвление}(V)=v$, что $v = (\alpha_1, \alpha_2, \dots, \alpha_k)$ и для любой вершины $v' = (\alpha'_1, \alpha'_2, \dots, \alpha'_k) \in V$ выполняется $k \leq k'$. Очевидно, что данная реализация метода ветвления будет соответствовать обходу дерева ветвления в ширину.

Оптимистичная стратегия. $\text{Ветвление}(V)=v$, что и для любой вершины $v' \in V$ выполняется $H(v) \leq H(v')$. Т.е. мы ветвимся в том направлении, в котором с точки зрения нижней оценки есть наибольшие перспективы улучшить значение критерия. Т.к. нижняя оценка является в общем случае недостижимой, то такая стратегия может быть охарактеризована как оптимистичная.

Реалистичная стратегия. $\text{Ветвление}(V)=v$, что и для любой вершины $v' \in V$ выполняется $B(v) \leq B(v')$. Т.е. мы ветвимся в том направлении, в котором с точки зрения верхней оценки есть наибольшие перспективы улучшить значение

критерия. Т.к. верхняя оценка достижима, то такая стратегия может быть охарактеризована как реалистичная.

Гибридная стратегия. Гибридная стратегия ветвления на основе комбинаций описанных выше стратегий.

Верхняя оценка

При поиске верхней оценки можно воспользоваться следующим жадным алгоритмом построения допустимого решения. Рассмотрим вершину $v = (\alpha_1, \alpha_2, \dots, \alpha_k)$, тогда обозначим $\{\beta_1, \beta_2, \dots, \beta_{n-k}\} = \{1, 2, \dots, n\} \setminus \{\alpha_1, \alpha_2, \dots, \alpha_k\}$.

Алгоритм 2.2. (верхняя оценка – жадная алгоритм)

Вход: $v = (\alpha_1, \alpha_2, \dots, \alpha_k)$.

Выход: $V(v)$.

Шаг 1. Пусть $x_i = \alpha_i$, $i = \overline{1, k}$. Если $k = n$, то алгоритм завершен; иначе $j = 0$, перейти на шаг 2.

Шаг 2. Обозначим $\{\beta_1, \beta_2, \dots, \beta_{n-(k+j)}\} = \{1, 2, \dots, n\} \setminus \{x_1, x_2, \dots, x_{k+j}\}$. Далее обозначим

$z = t_{0x_1} + \sum_{i=2}^{n-(k+j)} t_{x_{i-1}x_i}$. Пусть

$$w_{\beta_i} = \begin{cases} t_{\beta_i}^D - (z + t_{x_{k+j}\beta_i}), & \text{если } z + t_{x_{k+j}\beta_i} \leq t_{\beta_i}^D, i = \overline{1, n-(k+j)}. \\ +\infty, & \text{иначе} \end{cases}$$

Определим $h = \arg \min_{i=\overline{1, n-(k+j)}} w_{\beta_i}$. Тогда пусть $j = j+1$ и $x_{k+j} = \beta_h$. Если $k+j = n$, то алгоритм завершен; иначе перейти на шаг 2.

Содержательно предложенная верхняя оценка определяется как значение критерия, полученное для следующего решения. Изначально заказы выполняются в порядке $\alpha_1, \alpha_2, \dots, \alpha_k$. Далее каждый следующий заказ определяется как заказ, время возможного выполнения которого наиболее близко к его директивному сроку (если директивный срок при этом не нарушается).

Нижняя оценка

Как и ранее рассмотрим вершину $v = (\alpha_1, \alpha_2, \dots, \alpha_k)$ и обозначим $\{\beta_1, \beta_2, \dots, \beta_{n-k}\} = \{1, 2, \dots, n\} \setminus \{\alpha_1, \alpha_2, \dots, \alpha_k\}$. Далее обозначим

$$z_{\alpha_i} = t_{0\alpha_i} + \sum_{j=1}^{i-1} t_{\alpha_j\alpha_{j+1}}, i = \overline{1, k},$$

$$z_{\beta_i} = t_{0\alpha_1} + \sum_{j=1}^{k-1} t_{\alpha_j\alpha_{j+1}} + t_{\alpha_k\beta_i}, i = \overline{1, n-k},$$

$$w_i = \begin{cases} 0, & \text{если } z_i \leq t_i^D, \\ 1, & \text{иначе} \end{cases} i = \overline{1, n}.$$

Благодаря условию (2.1) в качестве нижней оценки может быть использована следующая величина:

$$H(v) = \sum_{i=1}^n w_i.$$

Содержательно данная нижняя оценка определяется следующим образом. Мы считаем, что изначально заказы выполняются в порядке $\alpha_1, \alpha_2, \dots, \alpha_k$. Далее из пункта доставки заказа α_k заказы $\beta_1, \beta_2, \dots, \beta_{n-k}$ выполняются «параллельно». Не сложно увидеть, что получаемое при этом число количество заказов, выполненных без нарушения директивных сроков, является нижней оценкой.

2.3. Задание

При выполнении лабораторной работы необходимо:

1. для каждой из задач оценить число допустимых решений N_D ,
2. реализовать метод ветвей и границ, используя описанные верхние и нижние оценки и стратегию ветвления в глубину (*базовый алгоритм*),
3. для каждой из тестовых задач оценить эффективность базового алгоритма как $(N_{\text{Виг}} - N_D) / N_D$, где $N_{\text{Виг}}$ – число обработанных вершин дерева ветвлений в выбранной реализации метода ветвей и границ,
4. оценить среднюю эффективность базового алгоритма на тестовых задачах,
5. предложить свою реализацию метода ветвей и границ (стратегии ветвления, верхней и нижней оценок), средняя эффективность которого превосходит базовым алгоритмом.

Требования к программной реализации:

1. метод ветвей и границ должен быть реализован в виде функции, принимающей в качестве параметров стратегию ветвления, алгоритм расчета нижней верхней оценки и алгоритм расчета нижней оценки,
2. стратегия ветвления должна принимать на вход текущее множество листьев дерева ветвлений и на выходе возвращать вершину для ветвления,
3. алгоритмы расчета верхней и нижней оценок должны принимать на вход вершину дерева ветвлений,
4. программная система должна предоставлять возможность запускать выбранный набор тестовых задач, используя метод ветвей и границ с выбранными стратегиями ветвления, алгоритмами построения верхней и нижней оценок,
5. в результате запуска программной системы необходимо сформировать таблицу эффективностей метода ветвей и границ для каждой из задач на каждой из реализаций метода ветвей и границ, в том числе средних значений эффективности по каждой из реализаций метода ветвей и границ,
6. программная система должна сохранять найденные решения тестовых задач.

2.4. Тестовые задачи

Формат описания исходных данных: число заказов n ; директивные сроки заказов t_i^D , $i = \overline{1, n}$; матрица длительностей перемещения t_{ij} , $i, j = \overline{0, n}$.

Задача 1.

$n=3$

директивные сроки: 26 29 28

матрица длительностей перемещения:

0	15	15	5
15	0	0	15
15	0	0	15
5	15	15	0

Оптимальный порядок доставки заказов: 3 1 2

Оптимальное значение критерия: 0

Задача 2.

$n=3$

директивные сроки: 19 23 19

матрица длительностей перемещения:

0	20	7	20
20	0	15	0
7	15	0	15
20	0	15	0

Оптимальный порядок доставки заказов: 2 1 3

Оптимальное значение критерия: 2

Задача 3.

$n=10$

директивные сроки: 129 33 99 82 123 98 84 38 48 75

матрица длительностей перемещения:

0	5	5	5	5	10	82	67	60	74	75
5	0	7	0	0	11	78	63	56	70	71
5	7	0	7	7	5	85	70	63	77	78
5	0	7	0	0	11	78	63	56	70	71
5	0	7	0	0	11	78	63	56	70	71
10	11	5	11	11	0	88	74	67	81	81
82	78	85	78	78	88	0	15	22	10	7
67	63	70	63	63	74	15	0	7	7	10
60	56	63	56	56	67	22	7	0	14	15
74	70	77	70	70	81	10	7	14	0	7
75	71	78	71	71	81	7	10	15	7	0

Оптимальный порядок доставки заказов: 1 3 2 4 7 6 5 8 9 10

Оптимальное значение критерия: 4

Задача 4.

$n=10$

директивные сроки: 49 117 112 103 115 97 92 51 119 119

матрица длительностей перемещения:

0	5	14	11	5	14	78	71	63	68	75
5	0	11	10	7	11	81	75	67	71	78
14	11	0	5	11	0	78	73	65	67	74
11	10	5	0	7	5	74	69	61	63	70
5	7	11	7	0	11	74	68	60	64	71

14 11 0 5 11 0 78 73 65 67 74
 78 81 78 74 74 78 0 15 15 11 5
 71 75 73 69 68 73 15 0 10 18 18
 63 67 65 61 60 65 15 10 0 11 15
 68 71 67 63 64 67 11 18 11 0 7
 75 78 74 70 71 74 5 18 15 7 0

Оптимальный порядок доставки заказов: 1 2 3 4 5 9 10 6 7 8
 Оптимальное значение критерия: 3

Задача 5.

$n=10$

директивные сроки: 113 110 109 106 123 59 82 96 63 58

матрица длительностей перемещения:

0 10 11 11 14 14 67 57 82 63 74
 10 0 11 11 10 10 75 63 88 71 81
 11 11 0 0 5 5 65 53 77 61 71
 11 11 0 0 5 5 65 53 77 61 71
 14 10 5 5 0 0 69 57 81 65 75
 14 10 5 5 0 0 69 57 81 65 75
 67 75 65 65 69 69 0 15 20 5 10
 57 63 53 53 57 57 15 0 25 10 18
 82 88 77 77 81 81 20 25 0 20 10
 63 71 61 61 65 65 5 10 20 0 11
 74 81 71 71 75 75 10 18 10 11 0

Оптимальный порядок доставки заказов: 1 4 5 2 3 7 6 8 9 10
 Оптимальное значение критерия: 4

Задача 6.

$n=15$

директивные сроки: 11 30 0 18 42 3 74 95 107 109 108 122 98 120 123

матрица длительностей перемещения:

0 4 4 4 5 4 32 28 29 30 14 12 12 11 14 13
 4 0 5 3 5 2 28 25 25 27 18 15 15 15 17 16
 4 5 0 2 1 3 30 27 28 29 15 14 14 13 16 15
 4 3 2 0 3 1 29 25 26 28 17 15 15 14 17 16
 5 5 1 3 0 4 30 26 27 29 16 15 14 14 17 16
 4 2 3 1 4 0 28 25 25 27 18 15 15 15 18 16
 32 28 30 29 30 28 0 4 4 4 45 43 43 42 45 44
 28 25 27 25 26 25 4 0 4 5 42 40 39 39 42 40
 29 25 28 26 27 25 4 4 0 2 42 40 40 39 42 41
 30 27 29 28 29 27 4 5 2 0 44 42 42 41 44 42
 14 18 15 17 16 18 45 42 42 44 0 5 3 4 3 5
 12 15 14 15 15 15 43 40 40 42 5 0 2 1 3 1
 12 15 14 15 14 15 43 39 40 42 3 2 0 1 3 3
 11 15 13 14 14 15 42 39 39 41 4 1 1 0 3 2
 14 17 16 17 17 18 45 42 42 44 3 3 3 3 0 2
 13 16 15 16 16 16 44 40 41 42 5 1 3 2 2 0

Оптимальный порядок доставки заказов: 1 2 4 5 9 7 8 13 11 10 14 12 15 3 6

Оптимальное значение критерия: 2

Задача 7.

$n=15$

директивные сроки: 32 28 27 33 49 15 63 83 84 70 77 93 85 114 115

матрица длительностей перемещения:

0	4	5	4	5	3	31	30	31	32	15	13	11	12	12	13
4	0	1	4	1	1	28	27	28	30	18	15	14	15	15	17
5	1	0	5	0	2	29	27	29	30	17	15	13	14	15	16
4	4	5	0	5	4	28	26	27	29	18	17	15	15	15	17
5	1	0	5	0	2	29	27	29	30	17	15	13	14	15	16
3	1	2	4	2	0	29	27	29	30	17	15	13	14	14	16
31	28	29	28	29	29	0	2	3	2	45	43	42	42	42	44
30	27	27	26	27	27	2	0	4	4	44	42	40	41	41	42
31	28	29	27	29	29	3	4	0	2	45	43	42	42	42	44
32	30	30	29	30	30	2	4	2	0	46	44	43	44	44	45
15	18	17	18	17	17	45	44	45	46	0	3	4	3	4	4
13	15	15	17	15	15	43	42	43	44	3	0	2	3	5	5
11	14	13	15	13	13	42	40	42	43	4	2	0	1	4	5
12	15	14	15	14	14	42	41	42	44	3	3	1	0	3	4
12	15	15	15	15	14	42	41	42	44	4	5	4	3	0	2
13	17	16	17	16	16	44	42	44	45	4	5	5	4	2	0

Оптимальный порядок доставки заказов: 1 2 4 3 5 12 13 11 10 15 14 8 9 6 7

Оптимальное значение критерия: 2

Задача 8.

$n=50$

директивные сроки: 6 9 24 29 13 15 12 39 6 30 27 32 17 1 4 43 35 489 465

425 426 447 419 488 449 463 431 436 424 499 461 400 498 988 747 931 727

757 718 733 919 730 951 892 871 772 951 957 834 972

матрица длительностей перемещения:

0 4 3 1 5 2 3 4 2 1 1 0 3 1 4 0 1 23 26 26 27 25 26 27 25 26 26 24 27 25 25 27 27 15 13 15 15 17 17 15 14 15 16 15 16 15 15 17 18 15 15
 4 0 1 3 1 4 4 2 3 4 3 4 4 3 2 4 4 25 28 27 28 26 27 28 27 27 28 25 29 26 26 28 29 14 12 14 14 15 15 14 13 14 15 14 14 13 14 15 17 14 14
 3 1 0 2 2 3 3 3 3 2 3 3 2 3 3 24 27 27 27 25 27 27 26 27 27 25 28 25 25 27 28 15 13 14 15 16 16 15 13 15 15 15 15 14 15 16 17 15 15
 1 3 2 0 4 1 4 5 3 2 2 1 4 2 4 1 2 22 25 25 26 24 25 26 25 25 25 23 27 24 24 26 27 16 14 15 16 17 17 16 15 16 17 16 16 15 16 17 19 16 16
 5 1 2 4 0 5 4 1 4 5 4 5 4 3 5 5 25 28 28 29 27 28 29 28 28 29 26 30 27 27 29 30 14 11 13 14 15 15 13 12 14 14 13 13 13 14 15 16 14 17
 2 4 3 1 5 0 5 4 3 3 2 5 3 5 2 3 22 25 25 23 25 25 24 25 25 22 26 23 23 25 26 17 15 16 17 18 18 17 15 17 18 17 16 17 18 20 17 14
 3 4 3 4 4 5 0 3 2 3 3 0 3 2 3 3 26 29 29 29 28 29 30 28 29 29 27 30 27 28 30 30 13 10 12 13 14 14 13 11 13 13 13 13 12 13 14 15 13 13
 4 2 3 5 1 5 3 0 4 5 3 4 3 3 2 4 5 26 29 29 30 27 29 29 28 29 29 27 30 27 27 29 30 13 10 12 13 14 14 13 11 13 14 13 13 12 13 14 15 13 13
 2 3 3 3 4 2 4 0 1 1 2 2 1 2 2 1 25 27 27 28 26 28 28 27 27 27 25 29 26 26 28 29 14 12 13 14 15 15 14 13 14 15 14 14 14 15 17 14 14
 1 4 3 2 5 3 3 5 1 0 2 1 3 2 3 1 0 24 27 27 27 25 27 28 26 27 27 25 28 25 28 28 15 13 14 15 16 16 15 14 15 15 15 15 14 15 16 18 15 15
 1 3 2 2 4 3 3 3 1 2 0 1 3 0 3 1 2 24 27 27 27 25 27 27 26 27 27 25 28 25 25 28 28 15 13 14 15 16 16 15 13 15 15 15 15 14 15 16 18 15 15
 0 4 3 1 5 2 3 4 2 1 1 0 3 1 4 0 1 23 26 26 27 25 26 27 25 26 26 24 27 25 25 27 27 15 13 15 15 17 17 15 14 15 16 15 16 15 15 17 18 15 15
 3 4 3 4 4 5 0 3 2 3 3 0 3 2 3 3 26 29 29 29 28 29 30 28 29 29 27 30 27 28 30 30 13 10 12 13 14 14 13 11 13 13 13 13 12 13 14 15 13 13
 1 3 2 2 4 3 3 3 1 2 0 1 3 0 3 1 2 24 27 27 27 25 27 27 26 27 27 25 28 25 28 28 15 13 14 15 16 16 15 13 15 15 15 15 14 15 16 18 15 15
 4 2 3 4 3 5 2 2 2 3 3 4 2 3 0 4 3 26 29 29 29 27 29 29 28 29 29 27 30 27 27 30 30 13 10 12 13 14 14 13 11 13 13 13 13 12 13 14 15 13 13
 0 4 3 1 5 2 3 4 2 1 1 0 3 1 4 0 1 23 26 26 27 25 26 27 25 26 26 24 27 25 25 27 27 15 13 15 15 17 17 15 14 15 16 15 16 15 15 17 18 15 15
 1 4 3 2 5 3 3 5 1 0 2 1 3 2 3 1 0 24 27 27 27 25 27 28 26 27 27 25 28 25 25 28 28 15 13 14 15 16 16 15 14 15 15 15 15 14 15 16 18 15 15
 23 25 24 22 25 22 26 26 25 24 24 23 26 24 26 23 24 0 4 3 4 2 4 4 3 3 4 1 5 2 2 5 5 38 36 37 38 39 39 38 37 38 39 38 38 37 38 39 41 38 38
 26 28 27 25 28 25 29 29 27 27 27 26 29 27 29 26 27 4 0 2 1 4 5 3 1 2 2 3 2 2 4 4 2 41 39 40 41 42 42 41 39 41 42 41 41 40 41 42 44 41 41
 26 27 27 25 28 25 29 29 27 27 27 26 29 27 29 26 27 3 2 0 1 2 3 1 3 0 3 3 2 2 3 2 41 39 40 41 42 42 41 39 41 42 41 41 40 41 42 44 41 41
 27 28 27 26 29 25 29 30 28 27 27 27 29 27 29 27 27 4 1 1 0 3 4 2 2 1 3 4 1 3 3 3 1 42 39 41 42 43 43 42 40 42 42 42 41 42 43 44 42 42
 25 26 25 24 27 23 28 27 26 25 25 28 25 27 25 25 2 4 2 3 0 2 3 4 2 5 1 4 2 0 3 4 40 37 39 40 41 41 39 38 40 40 39 39 39 40 41 42 39 40
 26 27 27 25 28 25 29 29 28 27 27 26 29 27 29 26 27 4 5 3 4 2 0 3 5 3 6 3 4 4 2 1 5 41 39 40 41 42 42 41 39 41 42 41 40 41 42 44 41 41
 27 28 27 26 29 25 30 29 28 28 27 27 30 27 29 27 28 4 3 1 2 3 3 0 4 1 4 3 1 3 3 2 3 42 39 41 42 43 43 42 40 42 42 42 42 41 42 43 44 42 42
 25 27 26 25 28 24 28 28 27 26 26 25 28 26 28 25 26 3 1 3 2 4 5 4 0 3 1 4 3 3 4 5 3 40 38 39 40 42 42 40 39 40 41 40 40 40 40 42 43 40 40
 26 27 27 25 28 25 29 29 27 27 27 26 29 27 29 26 27 3 2 0 1 2 3 1 3 0 3 3 2 2 3 2 41 39 40 41 42 42 41 39 41 42 41 41 40 41 42 44 41 41
 26 28 27 25 29 25 29 27 27 27 26 29 27 29 26 27 4 2 3 3 5 6 4 1 3 0 5 4 4 5 5 2 41 39 40 41 42 42 41 40 41 42 41 40 41 42 44 41 41
 24 25 25 23 26 22 27 27 25 25 25 24 27 25 27 24 25 1 3 3 4 1 3 3 4 3 5 0 4 1 1 4 5 39 37 38 39 40 40 39 37 39 39 39 39 38 39 40 42 39 39
 27 29 28 27 30 26 30 30 29 28 28 27 30 28 30 27 28 5 2 2 1 4 4 1 3 2 4 4 0 3 4 3 2 42 40 42 42 44 44 42 41 42 43 42 42 42 44 45 42 42
 25 26 25 24 27 23 27 27 26 25 25 25 27 25 27 25 25 2 2 2 3 2 4 3 3 2 4 1 3 0 2 3 4 39 37 39 39 41 41 39 38 39 40 39 39 39 41 42 39 39
 25 26 25 24 27 23 28 27 26 25 25 25 28 25 27 25 25 2 4 2 3 0 2 3 4 2 5 1 4 2 0 3 4 40 37 39 40 41 41 39 38 40 40 39 39 39 40 41 42 39 40
 27 28 27 26 29 25 30 29 28 28 28 27 30 28 30 27 28 5 4 3 3 3 1 2 5 3 5 4 3 3 3 0 4 42 39 41 42 43 43 42 40 42 42 42 42 41 42 43 44 42 42
 27 29 28 27 30 26 30 30 29 28 28 27 30 28 30 27 28 5 2 2 1 4 5 3 3 2 2 5 2 4 4 4 0 42 40 42 42 44 44 42 41 42 43 42 42 42 44 45 42 42
 15 14 15 16 14 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 40 41 42 40 41 41 39 42 39 40 42 42 0 3 1 0 2 2 3 4 0 1 3 5 4 0 2 4 2 0
 13 12 13 14 11 15 10 10 12 13 13 13 10 13 10 13 13 36 39 39 39 37 39 39 38 39 39 37 40 37 37 39 40 3 0 2 3 4 4 3 1 3 4 3 3 2 3 4 5 3 3
 15 14 14 15 13 16 12 12 13 14 14 15 12 14 12 15 14 37 40 40 41 39 40 41 39 40 40 38 42 39 39 41 42 1 2 0 1 3 3 3 3 1 2 3 4 3 1 3 5 1 1
 15 14 15 16 14 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 40 41 42 40 41 41 39 42 39 40 42 42 0 3 1 0 2 2 3 4 0 1 3 5 4 0 2 4 2 0
 17 15 16 17 15 18 14 14 15 16 16 17 14 16 14 17 16 39 42 42 43 41 42 43 42 42 42 40 44 41 41 43 44 2 4 3 2 0 0 2 4 2 1 2 4 3 2 0 2 2 2
 17 15 16 17 15 18 14 14 15 16 16 17 14 16 14 17 16 39 42 42 43 41 42 43 42 42 42 40 44 41 41 43 44 2 4 3 2 0 0 2 4 2 1 2 4 3 2 0 2 2 2
 15 14 15 16 13 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 39 41 42 40 41 41 39 42 39 39 42 42 3 3 3 3 2 2 0 2 3 3 0 2 1 3 2 3 2 3
 14 13 13 15 12 15 11 11 13 14 13 14 11 13 11 14 14 37 39 39 40 38 39 40 39 39 40 37 41 38 38 40 41 4 1 3 4 4 4 2 0 4 3 2 2 1 4 4 5 2 4
 15 14 15 16 14 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 40 41 42 40 41 42 40 41 41 39 42 39 40 42 42 0 3 1 0 2 2 3 4 0 1 3 5 4 0 2 4 2 0
 16 15 15 17 14 18 13 14 15 15 15 16 13 15 13 16 15 39 42 42 42 40 42 42 41 42 42 43 42 40 40 42 43 1 4 2 1 1 1 3 3 1 0 3 4 0 1 3 4 0 1 3 1 1
 15 14 15 16 13 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 39 41 42 40 41 41 39 42 39 39 42 42 3 3 3 3 2 2 0 2 3 3 0 2 1 3 2 3 2 3
 16 14 15 16 13 17 13 13 14 15 15 16 13 15 13 16 15 38 41 41 42 39 41 42 40 41 41 39 42 39 39 42 42 5 3 4 5 4 4 2 2 5 4 2 0 1 5 4 4 3 5
 15 13 14 15 13 16 12 12 14 14 14 15 12 14 12 15 14 37 40 40 41 39 40 41 40 40 40 38 42 39 39 41 42 4 2 3 4 3 3 1 1 4 4 1 1 0 4 3 4 3 4
 15 14 15 16 14 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 40 41 42 40 41 41 39 42 39 40 42 42 0 3 1 0 2 2 3 4 0 1 3 5 4 0 2 4 2 0
 17 15 16 17 15 18 14 14 15 16 16 17 14 16 14 17 16 39 42 42 43 41 42 43 42 42 42 40 44 41 41 43 44 2 4 3 2 0 0 2 4 2 1 2 4 3 2 0 2 2 2
 18 17 17 19 16 20 15 15 17 18 18 18 15 18 15 18 18 41 44 44 44 42 44 44 43 44 44 42 45 42 42 44 45 4 5 5 4 2 2 3 5 4 3 3 4 4 4 2 0 4 4
 15 14 15 16 14 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 39 41 42 40 41 41 39 42 39 39 42 42 2 3 1 2 2 2 2 2 2 1 2 3 3 2 2 4 0 2
 15 14 15 16 14 17 13 13 14 15 15 15 13 15 13 15 15 38 41 41 42 40 41 42 40 41 41 39 42 39 40 42 42 0 3 1 0 2 2 3 4 0 1 3 5 4 0 2 4 2 0

Оптимальный порядок доставки заказов: 15 16 9 2 1 8 13 10 7 6 3 5 11 4 12 32 17 23
29 20 21 27 28 22 25 31 26 19 24 18 33 30 39 37 42 40 35 38 46 49 45 44 41 36 43 47 48
50 34 14

Оптимальное значение критерия: 3

Задача 9.

$n=50$

директивные сроки: 37 39 5 5 30 31 28 40 43 11 18 29 33 4 26 49 7 448 406
415 498 403 433 403 451 457 485 422 411 415 477 483 475 772 752 737 753
873 918 948 768 820 713 784 845 807 882 824 938 757

матрица длительностей перемещения:

0 4 4 2 2 4 5 3 5 5 5 4 5 4 2 5 5 30 31 32 30 32 30 30 33 29 29 29 29 32 31 30 28 10 10 11 12 10 13 15 10 12 10 12 12 14 15 12 12 10 9
4 0 1 3 3 3 1 1 3 3 3 3 4 3 3 1 26 27 28 26 29 27 27 29 25 26 26 25 29 27 27 25 13 13 15 15 14 17 18 13 15 14 16 15 18 18 15 15 14 13
4 1 0 4 4 3 2 2 4 4 4 3 4 5 4 4 2 27 28 29 27 30 27 28 30 26 27 27 26 30 28 27 25 13 13 14 15 13 16 18 13 15 13 15 15 17 18 15 15 13 12
2 3 4 0 0 2 4 2 3 3 3 2 3 2 0 3 4 28 30 30 28 31 29 29 32 27 28 28 27 31 29 29 27 11 11 13 13 12 15 16 11 14 12 13 13 15 16 13 13 12 10
2 3 4 0 0 2 4 2 3 3 2 3 2 0 3 4 28 30 30 28 31 29 29 32 27 28 28 27 31 29 29 27 11 11 13 13 12 15 16 11 14 12 13 13 15 16 13 13 12 10
4 3 3 2 2 0 2 2 2 2 2 2 2 2 2 4 27 28 29 27 29 28 27 30 26 27 27 26 29 28 27 25 13 13 14 15 13 16 18 13 15 13 15 15 17 18 15 15 13 12
5 1 2 4 4 2 0 2 2 2 2 2 4 4 2 2 25 27 27 25 28 26 26 29 25 25 25 25 28 27 26 24 14 14 15 16 15 18 19 14 16 15 16 16 18 19 16 16 15 13
3 1 2 2 2 2 0 2 2 2 2 2 3 2 2 2 27 28 29 27 29 28 27 30 26 27 27 26 29 28 27 25 13 13 14 15 13 16 17 13 15 13 15 15 17 17 15 15 13 12
5 3 4 3 3 2 2 2 0 0 2 0 2 3 0 3 25 27 27 25 28 26 26 29 25 25 25 25 28 27 26 24 14 14 16 16 15 17 19 14 16 15 16 16 18 19 16 16 15 13
5 3 4 3 3 2 2 2 0 0 2 0 2 3 0 3 25 27 27 25 28 26 26 29 25 25 25 25 28 27 26 24 14 14 16 16 15 17 19 14 16 15 16 16 18 19 16 16 15 13
5 3 4 3 3 2 2 2 0 0 2 0 2 3 0 3 25 27 27 25 28 26 26 29 25 25 25 25 28 27 26 24 14 14 16 16 15 17 19 14 16 15 16 16 18 19 16 16 15 13
4 3 3 2 2 0 2 2 2 2 2 0 2 2 2 2 4 27 28 29 27 29 28 27 30 26 27 27 26 29 28 27 25 13 13 14 15 13 16 18 13 15 13 15 15 17 18 15 15 13 12
5 3 4 3 3 2 2 2 0 0 2 0 2 3 0 3 25 27 27 25 28 26 26 29 25 25 25 25 28 27 26 24 14 14 16 16 15 17 19 14 16 15 16 16 18 19 16 16 15 13
4 4 5 2 2 4 3 2 2 2 2 0 2 2 5 27 29 29 27 30 28 27 30 26 27 27 26 30 28 28 25 13 13 15 15 14 16 18 13 15 14 15 15 17 18 15 15 14 12
2 3 4 0 0 2 4 2 3 3 3 2 3 2 0 3 4 28 30 30 28 31 29 29 32 27 28 28 27 31 29 29 27 11 11 13 13 12 15 16 11 14 12 13 13 15 16 13 13 12 10
5 3 4 3 3 2 2 2 0 0 2 0 2 3 0 3 25 27 27 25 28 26 26 29 25 25 25 25 28 27 26 24 14 14 16 16 15 17 19 14 16 15 16 16 18 19 16 16 15 13
5 1 2 4 4 2 2 2 3 3 4 3 5 4 3 0 25 27 28 25 28 26 26 29 25 25 25 25 28 27 26 24 14 14 15 16 15 18 19 14 16 15 17 16 18 19 16 16 15 13
30 26 27 28 28 27 25 25 25 27 25 27 28 25 25 0 2 5 5 4 1 4 5 4 3 2 3 4 4 3 2 3 9 39 40 41 39 42 44 39 41 39 41 41 43 44 41 41 39 38
31 27 28 30 30 28 27 28 27 27 27 28 27 29 30 27 27 2 0 4 5 4 1 4 3 5 4 2 3 4 3 3 4 40 40 42 42 41 44 45 40 42 41 43 42 44 45 42 42 41 39
32 28 29 30 30 29 27 29 27 27 27 29 27 29 30 27 28 5 4 0 3 1 5 2 2 3 3 3 4 1 1 2 5 41 41 42 43 42 44 46 41 43 42 43 43 45 46 43 43 42 40
30 26 27 28 28 27 25 27 25 25 25 27 25 27 28 25 25 5 5 3 0 4 5 1 5 1 2 3 3 4 2 3 4 39 39 40 41 39 42 44 39 41 39 41 41 43 44 41 41 39 38
32 29 30 31 31 29 28 28 28 28 28 28 29 28 30 31 28 28 4 4 1 4 0 5 3 1 4 3 4 4 0 2 3 5 42 42 43 44 42 45 46 42 44 42 44 44 46 44 42 41
30 27 27 29 29 28 26 28 26 26 26 28 28 28 29 26 26 1 1 5 5 5 0 5 4 5 4 3 4 5 4 3 3 40 39 41 42 40 43 44 40 42 40 42 42 44 42 40 39
30 27 28 29 29 27 26 27 26 26 26 27 26 27 29 26 26 4 4 2 1 3 5 0 4 2 1 3 2 3 1 2 3 39 39 41 42 40 43 44 39 42 40 42 42 44 44 42 42 40 39
33 29 30 32 32 30 29 30 29 29 29 30 29 30 32 29 29 5 3 2 5 1 4 4 0 5 4 4 5 1 3 3 5 42 42 44 44 43 46 47 42 44 43 44 44 46 47 44 44 43 42
29 25 26 27 27 26 25 26 25 25 25 26 25 26 27 25 25 4 5 3 1 4 5 2 5 0 1 3 2 4 3 2 3 38 38 40 40 39 42 43 38 40 39 40 40 42 43 40 40 39 37
29 26 27 28 28 27 25 27 25 25 25 27 25 27 28 25 25 3 4 3 2 3 4 1 4 1 0 2 1 3 2 1 2 39 39 40 41 39 42 44 39 41 39 41 41 43 44 41 41 39 38
29 26 27 28 28 27 25 27 25 25 25 27 25 27 28 25 25 2 2 3 3 4 3 3 4 3 2 0 1 4 2 1 2 39 39 40 41 39 42 44 39 41 39 41 41 43 44 41 41 39 38
29 25 26 27 27 26 25 26 25 25 25 26 25 26 27 25 25 3 3 4 3 4 4 2 5 2 1 1 0 4 3 2 1 38 38 39 40 39 42 43 38 40 39 40 40 42 43 40 40 39 37
32 29 30 31 31 29 28 28 28 28 28 29 28 30 31 28 28 4 4 1 4 0 5 3 1 4 3 4 4 0 2 3 5 42 42 43 44 42 45 46 42 44 42 44 44 46 46 44 44 42 41
31 27 28 29 29 28 27 28 27 27 27 28 27 28 29 27 27 4 3 1 2 2 4 1 3 3 2 2 0 1 4 40 40 42 42 41 44 45 40 42 41 42 42 44 45 42 41 41 39
30 27 27 29 29 27 26 27 26 26 26 27 26 28 29 26 26 3 3 2 3 3 3 2 3 2 1 1 2 3 1 0 3 39 39 41 42 40 43 44 39 42 40 42 42 44 44 42 42 40 39
28 25 25 27 27 25 24 25 24 24 24 25 24 25 27 24 24 2 4 5 4 5 3 3 5 3 2 2 1 5 4 3 0 37 37 39 39 38 41 42 37 39 38 40 39 42 42 39 38 37
10 13 13 11 11 13 14 13 14 14 14 13 14 13 11 14 14 39 40 41 39 42 40 39 42 38 39 39 38 42 40 39 37 0 2 4 3 1 4 5 0 3 1 3 3 5 5 3 3 1 1
10 13 13 11 11 13 14 13 14 14 14 13 14 13 11 14 14 39 40 41 39 42 39 39 42 38 39 39 38 42 40 39 37 2 0 2 3 1 5 5 2 3 1 5 3 5 5 3 3 1 1
11 15 14 13 13 14 15 14 16 16 16 14 16 15 13 16 15 40 42 42 40 43 41 41 44 40 40 40 39 43 42 41 39 4 2 0 3 3 5 5 4 1 3 5 3 4 5 4 3 3 3
12 15 15 13 13 15 16 15 16 16 16 15 16 15 13 16 16 41 42 43 41 44 42 42 44 40 41 41 40 44 42 42 39 3 3 3 0 2 2 3 3 2 2 3 0 3 3 2 0 2 3
10 14 13 12 12 13 15 13 15 15 15 13 15 14 12 15 15 39 41 42 39 42 40 40 43 39 39 39 42 41 40 38 1 1 3 2 0 4 5 1 2 0 4 2 4 5 2 2 0 2
13 17 16 15 15 16 18 16 17 17 17 16 17 16 15 17 18 42 44 44 42 45 43 43 46 42 42 42 42 45 44 43 41 4 5 5 2 4 0 2 4 4 2 2 1 2 2 2 4 5
15 18 18 16 16 18 19 17 19 19 19 18 19 18 16 19 19 44 45 46 44 46 44 44 47 43 44 44 43 46 45 44 42 5 5 5 3 5 2 0 5 4 5 4 3 1 0 4 3 5 6
10 13 13 11 11 13 14 13 14 14 14 13 14 13 11 14 14 39 40 41 39 42 40 39 42 38 39 39 38 42 40 39 37 0 2 4 3 1 4 5 0 3 1 3 3 5 5 3 3 1 1
12 15 15 14 14 15 16 15 16 16 16 15 16 15 14 16 16 41 42 43 41 44 42 42 44 40 41 41 40 44 42 42 39 3 3 3 0 2 2 3 3 2 2 3 0 3 3 2 0 2 4
10 14 13 12 12 13 15 13 15 15 15 13 15 14 12 15 15 39 41 42 39 42 40 40 43 39 39 39 42 41 40 38 1 1 3 2 0 4 5 1 2 0 4 2 4 5 2 2 0 2
13 17 16 15 15 16 18 16 17 17 17 16 17 16 15 17 18 42 44 44 42 45 43 43 46 42 42 42 42 45 44 43 41 4 5 5 2 4 0 2 4 4 2 2 1 2 2 2 4 5
15 18 18 16 16 18 19 17 19 19 19 18 19 18 16 19 19 44 45 46 44 46 44 44 47 43 44 44 43 46 45 44 42 5 5 5 3 5 2 0 5 4 5 4 3 1 0 4 3 5 6
10 13 13 11 11 13 14 13 14 14 14 13 14 13 11 14 14 39 40 41 39 42 40 39 42 38 39 39 38 42 40 39 37 0 2 4 3 1 4 5 0 3 1 3 3 5 5 3 3 1 1
12 15 15 14 14 15 16 15 16 16 16 15 16 15 14 16 16 41 42 43 41 44 42 42 44 40 41 41 40 44 42 42 39 3 3 1 2 2 4 4 3 0 2 5 2 3 4 3 2 2 4
10 14 13 12 12 13 15 13 15 15 15 13 15 14 12 15 15 39 41 42 39 42 40 40 43 39 39 39 42 41 40 38 1 1 3 2 0 4 5 1 2 0 4 5 1 2 0 4 2 4 5 2 2 0 2
12 16 15 13 13 15 16 15 16 16 16 15 16 15 13 16 17 41 43 43 41 44 42 42 44 40 41 41 40 44 42 42 40 3 5 5 3 4 2 4 3 5 4 0 3 3 4 2 3 4 4
12 15 15 13 13 15 16 15 16 16 16 15 16 15 13 16 16 41 42 43 41 44 42 42 44 40 41 41 40 44 42 42 39 3 3 3 0 2 2 3 3 2 2 3 0 3 3 2 0 2 3
14 18 17 15 15 17 18 17 18 18 18 17 18 17 15 18 18 43 44 45 43 46 44 44 46 42 43 43 42 46 44 44 42 5 5 4 3 4 1 1 5 3 4 3 3 0 1 3 3 4 5
15 18 18 16 16 18 19 17 19 19 19 18 19 18 16 19 19 44 45 46 44 46 44 44 47 43 44 44 43 46 45 44 42 5 5 5 3 5 2 0 5 4 5 4 3 1 0 4 3 5 6
12 15 15 13 13 15 16 15 16 16 16 15 16 15 13 16 16 41 42 43 41 44 42 42 44 40 41 41 40 44 42 42 39 3 3 4 2 2 2 4 3 3 2 2 2 3 4 0 2 2 4
12 15 15 13 13 15 16 15 16 16 16 15 16 15 13 16 16 41 42 43 41 44 42 42 44 40 41 41 40 44 42 42 39 3 3 3 0 2 2 3 3 2 2 3 0 3 3 2 0 2 3
10 14 13 12 12 13 15 13 15 15 15 13 15 14 12 15 15 39 41 42 39 42 40 40 43 39 39 39 42 41 40 38 1 1 3 2 0 4 5 1 2 0 4 2 4 5 2 2 0 2
9 13 12 10 10 12 13 13 13 12 13 12 10 13 13 38 39 40 38 41 39 39 42 37 38 38 37 41 39 39 37 1 1 3 3 2 5 6 1 4 2 4 3 5 6 4 3 2 0

Оптимальный порядок доставки заказов: 3 14 4 10 11 15 7 12 5 6 13 1 2 8 9 16 22 17
24 19 29 20 30 28 23 18 25 26 33 31 32 27 21 43 36 35 37 50 41 34 44 46 42 48 45 38 47
39 49 40

Оптимальное значение критерия: 1

Задача 10.

$n=50$

директивные сроки: 34 49 35 34 31 2 32 32 46 16 5 39 45 3 8 46 39 429 440
487 423 411 456 421 467 448 472 448 416 402 444 446 407 987 990 773 971
988 746 962 821 966 880 980 972 812 750 895 727 889

матрица длительностей перемещения:

0 3 1 2 4 3 4 3 3 5 5 4 1 3 2 3 3 25 28 27 29 27 27 28 25 27 26 30 27 27 27 27 28 13 13 14 13 16 15 14 12 16 15 15 14 14 14 17 16 17 15
3 0 3 3 1 0 5 5 5 5 5 3 2 4 1 0 2 27 30 29 31 30 29 30 27 29 28 32 29 29 29 29 30 10 11 12 11 14 13 12 10 14 13 13 12 12 12 15 14 15 13
1 3 0 1 4 3 3 2 2 4 4 3 2 2 3 3 2 25 27 26 28 27 26 27 25 27 25 29 26 27 26 26 27 13 14 15 14 17 15 15 13 17 16 16 15 15 17 17 18 15
2 3 1 0 3 3 2 3 3 3 3 2 1 1 2 3 1 25 28 27 29 28 27 27 25 27 26 29 27 27 27 28 13 13 14 13 16 15 14 12 16 15 15 14 14 14 17 16 17 15
4 1 4 3 0 1 5 5 5 4 4 2 3 4 2 1 3 28 30 29 32 30 29 30 28 30 29 32 30 30 30 31 10 10 11 10 13 12 11 9 13 13 13 11 11 11 14 13 14 12
3 0 3 3 1 0 5 5 5 5 5 3 2 4 1 0 2 27 30 29 31 30 29 30 27 29 28 32 29 29 29 29 30 10 11 12 11 14 13 12 10 14 13 13 12 10 14 13 15 14 15 13
4 5 3 2 5 5 0 3 1 2 2 3 3 1 4 5 3 24 26 25 28 26 25 26 24 26 25 28 25 26 25 25 27 14 15 15 15 17 17 16 13 18 17 17 16 16 15 18 18 18 16
3 5 2 3 5 5 3 0 2 5 5 5 3 2 4 5 4 23 25 25 27 25 25 25 23 25 24 27 25 25 25 25 26 15 15 16 15 18 17 16 14 18 18 17 16 16 19 18 19 17
3 5 2 3 5 5 1 2 0 3 3 4 4 2 4 5 3 23 25 25 27 25 25 25 23 25 24 27 25 25 25 25 26 15 15 16 15 18 17 16 14 18 18 18 16 16 19 18 19 17
5 5 4 3 4 5 2 5 3 0 0 2 4 3 4 5 3 25 27 27 29 28 27 27 25 28 26 30 27 28 27 27 29 13 14 14 14 16 16 15 12 17 16 16 15 15 14 17 16 17 15
5 5 4 3 4 5 2 5 3 0 0 2 4 3 4 5 3 25 27 27 29 28 27 27 25 28 26 30 27 28 27 27 29 13 14 14 14 16 16 15 12 17 16 16 15 15 14 17 16 17 15
4 3 3 2 2 3 3 5 4 2 2 0 3 3 2 3 1 27 29 28 30 29 28 29 27 29 28 31 28 29 28 28 30 11 12 13 12 15 14 13 10 15 14 14 13 13 13 15 15 15 13
1 2 2 1 3 2 3 3 4 4 4 3 0 2 1 2 2 26 28 27 29 28 27 28 26 28 27 30 28 28 28 28 29 12 13 13 13 15 14 13 11 15 15 15 13 13 14 16 15 16 14
3 4 2 1 4 4 1 2 2 3 3 2 0 3 4 2 25 27 26 28 27 26 27 25 27 25 29 26 27 26 26 28 13 14 15 14 17 16 15 13 17 16 16 15 15 18 17 17 15
2 1 3 2 2 1 4 4 4 4 4 2 1 3 0 1 1 27 29 28 30 29 28 29 27 29 27 31 28 29 28 28 30 11 12 13 12 15 14 13 10 15 14 14 13 13 15 15 15 13
3 0 3 3 1 0 5 5 5 5 5 3 2 4 1 0 2 27 30 29 31 30 29 30 27 29 28 32 29 29 29 29 30 10 11 12 11 14 13 12 10 14 13 13 12 12 12 15 14 15 13
3 2 2 1 3 2 3 4 3 3 3 1 2 2 1 2 0 26 28 27 30 28 27 28 26 28 27 30 27 28 27 27 29 12 13 13 13 15 14 14 11 15 15 15 14 14 13 16 15 16 14
25 27 25 25 28 27 24 23 23 25 25 27 26 25 27 27 26 0 5 2 4 3 2 3 2 3 1 5 4 3 4 4 4 37 38 39 38 41 39 39 37 41 40 40 39 39 42 41 42 39
28 30 27 28 30 30 26 25 25 27 27 29 28 27 29 30 28 5 0 3 5 6 4 2 3 5 5 3 1 5 1 1 5 39 40 41 40 43 42 41 39 43 43 42 41 41 41 44 43 44 42
27 29 26 27 29 29 25 25 25 27 27 28 27 26 28 29 27 2 3 0 3 4 2 1 2 3 3 3 2 3 2 2 4 39 39 40 39 42 41 40 38 42 42 42 40 40 40 43 42 43 41
29 31 28 29 32 31 28 27 27 29 29 30 29 28 30 31 30 4 5 3 0 2 3 4 5 2 3 3 5 2 5 5 1 41 42 42 42 44 43 42 40 44 44 44 42 42 42 45 44 45 43
27 30 27 28 30 30 26 25 25 28 28 29 28 27 29 30 28 3 6 4 2 0 3 5 5 2 2 5 5 2 5 5 1 40 40 41 40 43 42 41 39 43 42 42 41 41 44 43 44 42
27 29 26 27 29 29 25 25 25 27 27 28 27 26 28 29 27 2 4 2 3 3 0 3 2 1 1 4 3 1 3 3 2 39 39 40 39 42 41 40 38 42 42 42 40 40 40 43 42 43 41
28 30 27 27 30 30 26 25 25 27 27 29 28 27 29 30 28 3 2 1 4 5 3 0 3 3 4 3 1 3 1 1 4 39 40 41 40 43 42 41 39 43 42 42 41 41 41 44 43 44 42
25 27 25 25 28 27 24 23 23 25 25 27 26 25 27 27 26 2 3 2 5 5 2 3 0 3 3 5 2 3 2 2 4 37 38 39 38 41 40 39 37 41 40 40 39 39 39 42 41 42 39
27 29 27 27 30 29 26 25 25 28 28 29 28 27 29 29 28 3 5 3 2 2 1 3 3 0 2 3 4 0 4 4 1 39 40 41 40 43 42 41 39 43 42 42 41 41 41 44 43 44 42
26 28 25 26 29 28 25 24 24 26 26 28 27 25 27 28 27 1 5 3 3 2 1 4 3 2 0 5 4 2 4 4 3 38 39 39 39 42 40 39 37 42 41 41 39 40 42 42 42 40
30 32 29 29 32 32 28 27 27 30 30 31 30 29 31 32 30 5 3 3 3 5 4 3 5 3 5 0 4 3 4 4 4 42 42 43 42 45 44 43 41 45 44 44 43 43 43 46 45 46 44
27 29 26 27 30 29 25 25 25 27 27 28 28 26 28 29 27 4 1 2 5 5 3 1 2 4 4 4 0 4 0 0 5 39 39 40 39 42 41 40 38 42 42 42 40 40 40 43 42 43 41
27 29 27 27 30 29 26 25 25 28 28 29 28 27 29 29 28 3 5 3 2 2 1 3 3 0 2 3 4 0 4 4 1 39 40 41 40 43 42 41 39 43 42 42 41 41 41 44 43 44 42
27 29 26 27 30 29 25 25 25 27 27 28 28 26 28 29 27 4 1 2 5 3 1 2 4 4 0 4 0 0 5 39 39 40 39 42 41 40 38 42 42 42 40 40 40 43 42 43 41
27 29 26 27 30 29 25 25 25 27 27 28 28 26 28 29 27 4 1 2 5 5 3 1 2 4 4 4 0 4 0 0 5 39 39 40 39 42 41 40 38 42 42 42 40 40 40 43 42 43 41
28 30 27 28 31 30 27 26 26 29 29 30 29 28 30 30 29 4 5 4 1 1 2 4 4 1 3 4 5 1 5 5 0 40 41 42 41 44 42 42 39 44 43 43 42 42 42 44 44 44 42
13 10 13 13 10 10 14 15 15 13 13 11 12 13 11 10 12 37 39 39 41 40 39 39 37 39 38 42 39 39 39 39 40 0 1 2 1 4 5 4 1 5 4 4 4 4 2 5 4 5 3
13 11 14 13 10 11 15 15 15 14 14 12 13 14 12 11 13 38 40 39 42 40 39 40 38 40 39 42 39 40 39 39 41 1 0 1 0 4 4 3 2 4 3 3 3 3 4 3 4 2
14 12 15 14 11 12 15 16 16 14 14 13 13 15 13 12 13 39 41 40 42 41 40 41 39 41 39 43 40 41 40 40 42 2 1 0 1 3 4 3 3 3 4 2 3 3 2 4 3 3 1
13 11 14 13 10 11 15 15 15 14 14 12 13 14 12 11 13 38 40 39 42 40 39 40 38 40 39 42 39 40 39 39 41 1 0 1 0 4 4 3 2 4 3 3 3 3 4 3 4 2
16 14 17 16 13 14 17 18 18 16 16 15 15 17 15 14 15 41 43 42 44 43 42 43 41 43 42 45 42 43 42 42 44 4 4 3 4 0 5 5 5 3 4 3 5 5 3 2 1 2
15 13 15 15 12 13 17 17 17 16 16 14 14 16 14 13 14 39 42 41 43 42 41 42 40 42 40 44 41 42 41 41 42 5 4 4 4 5 0 1 4 2 1 3 1 1 5 3 4 5 3
14 12 15 14 11 12 16 16 16 15 15 13 13 15 13 12 14 39 41 40 42 41 40 41 39 41 39 43 40 41 40 40 42 4 3 3 3 5 1 0 3 3 2 2 0 0 5 4 3 4 3
12 10 13 12 9 10 13 14 14 12 12 10 11 13 10 10 11 37 39 38 40 39 38 39 37 39 37 41 38 39 38 38 39 1 2 3 2 5 4 3 0 5 5 4 3 3 3 5 5 5 3
16 14 17 16 13 14 18 18 18 17 17 15 15 17 15 14 15 41 43 42 44 43 42 43 41 43 42 45 42 43 42 42 44 5 4 3 4 3 2 3 5 0 1 1 3 3 5 1 2 3 2
15 13 16 15 13 13 17 17 18 16 16 14 15 16 14 13 15 40 42 42 44 42 42 42 40 42 41 44 42 42 42 42 43 4 3 2 3 3 2 4 1 2 0 2 2 4 2 1 2 1
14 12 15 14 11 12 16 16 16 15 15 13 13 15 13 12 14 39 41 40 42 41 40 41 39 41 39 43 40 41 40 40 42 4 3 3 3 5 1 0 3 3 2 2 0 0 5 4 3 4 3
14 12 15 14 11 12 16 16 16 15 15 13 13 15 13 12 14 39 41 40 42 41 40 41 39 41 39 43 40 41 40 40 42 4 3 3 3 5 1 0 3 3 2 2 0 0 5 4 3 4 3
14 12 15 14 11 12 15 16 16 14 14 13 14 15 13 12 13 39 41 40 42 41 40 41 39 41 39 43 40 41 40 40 42 2 3 2 3 3 5 3 5 5 4 5 5 0 4 3 4 3
17 15 17 17 14 15 18 19 19 17 17 15 16 18 15 15 16 42 44 43 45 44 43 44 42 44 42 46 43 44 43 43 44 5 4 4 4 3 3 4 5 1 2 2 4 4 4 0 1 2 3
16 14 17 16 13 14 18 18 18 16 16 15 15 17 15 14 15 41 43 42 44 43 42 43 41 43 42 45 42 43 42 42 44 4 3 3 3 2 4 3 5 2 3 1 3 3 3 1 0 1 2
17 15 18 17 14 15 18 19 19 17 17 15 16 17 15 15 16 42 44 43 45 44 43 44 42 44 42 46 43 44 43 43 44 5 4 3 4 1 5 4 5 3 4 2 4 4 4 2 1 0 3
15 13 15 15 12 13 16 17 17 15 15 13 14 15 13 13 14 39 42 41 43 42 41 42 39 42 40 44 41 42 41 41 42 3 2 1 2 2 3 3 3 2 3 1 3 3 3 2 3 0

Оптимальный порядок доставки заказов: 14 16 11 15 13 10 9 1 5 7 8 4 3 12 2 30 17 33
22 29 24 21 18 19 31 32 26 28 23 25 27 20 49 39 47 36 46 41 43 50 48 40 42 37 45 44 34
38 35 6

Оптимальное значение критерия: 2

Литература

1. Гэри М., Джонсон Д.. Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982.
2. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. М.: Мир, 1985.